L Number	Hits	Search Text	DB	Time stamp
-	150	data with lock$4 same dead$lock	USPAT; EPO; JPO	2004/07/22 10:28
-	83	data with lock$4 same dead$lock and data with request and data with loca$8	USPAT; EPO; JPO	2004/06/28 14:52
-	35	data with lock$4 same dead$lock and data with request and data with loca$8 and (709/$.ccls 707/$.ccls. 705/$.ccls.)	USPAT; EPO; JPO	2004/06/28 14:55
-	25	data with lock$4 same dead$lock and data with request and data with loca$8 and (709/$.ccls 707/$.ccls. 705/$.ccls.) not ginter.in.	USPAT; EPO; JPO	2004/06/28 15:33
-	1380	request with data same lock$4 with data (loca$8 area level) same dead$lock with history	USPAT; EPO; JPO	2004/06/28 15:06
-	0	request$4 with data same lock$4 with data with (loca$8 area level) same dead$lock with history	USPAT; EPO; JPO	2004/06/28 15:08
-	5	dead$lock with history	USPAT; EPO; JPO	2004/06/28 15:16
-	0	request$4 with data with access same dead$lock same lock$4 with database	USPAT; EPO; JPO	2004/06/28 16:02
-	34	request$4 with data with access same dead$lock	USPAT; EPO; JPO	2004/06/28 16:42
-	189	deadlock with (history record table)	USPAT; EPO; JPO	2004/06/28 16:55
-	53	access$4 with (history record table) same deadlock	USPAT; EPO; JPO	2004/06/28 16:57
-	4	access$4 with (history record table) same deadlock same lock$4 with data	USPAT; EPO; JPO	2004/06/29 09:59
-	2390	data with lock$4 with (level section area)	USPAT; EPO; JPO	2004/06/29 10:27
-	9	data with lock$4 with (level section area) same deadlock	USPAT; EPO; JPO	2004/06/29 10:47
-	230	dead$lock with (list table history log logging monitor$4)	USPAT; EPO; JPO	2004/06/29 10:48
-	68	dead$lock with (list table history log logging monitor$4) and lock$4 with (data file object section location area row table) same deadlock	USPAT; EPO; JPO	2004/06/29 10:49
-	4	lock$4 with data with level with history	USPAT; US-PGPUB; EPO; JPO	2004/08/02 10:40
-	48	lock$4 with data with level with (log past previous prior)	USPAT; US-PGPUB; EPO; JPO	2004/08/02 10:41
-	48	lock$4 with data with level with (log past previous prior) not (lock$4 with data with level with history)	USPAT; US-PGPUB; EPO; JPO	2004/08/02 11:27
-	272	dead$lock with (log history record past previous prior)	USPAT; US-PGPUB; EPO; JPO	2004/08/02 11:28
-	40	dead$lock with (log history record past previous prior) same lock$4 with (file data block portion section)	USPAT; US-PGPUB; EPO; JPO	2004/08/02 11:49
-	2	dead$lock with (log history record past previous prior) same (period time) with ("with" without) with (dead$lock)	USPAT; US-PGPUB; EPO; JPO	2004/08/02 12:12
-	1	dead$lock adj history	USPAT; US-PGPUB; EPO; JPO	2004/08/02 12:13
-	2	dead$lock adj (history record log)	USPAT; US-PGPUB; EPO; JPO	2004/08/02 13:14
-	4	dead$lock near (history record log)	USPAT; US-PGPUB; EPO; JPO	2004/08/02 13:14
-	2	dead$lock near (history record log) not (dead$lock adj (history record log))	USPAT; US-PGPUB; EPO; JPO	2004/08/02 13:14

L Number	Hits	Search Text	DB	Time stamp
-	3872	lock$4 with (history log record)	USPAT; EPO; JPO	2004/08/03 15:30
-	76	dead$lock$4 with (history log record)	USPAT; EPO; JPO	2004/08/03 15:30
-	85	dead$lock$4 with (history log record)	USPAT; EPO; JPO; DERWENT; IBM_TDB	2004/08/03 15:30
-	27	dead$lock$4 with (history log record) same lock	USPAT; EPO; JPO; DERWENT; IBM_TDB	2004/08/03 13:08
-	2	dead$lock$4 near2 (history log record) same lock	USPAT; EPO; JPO; DERWENT; IBM_TDB	2004/08/03 13:11
-	3	dead$lock$4 near2 (history past previous log record) same lock	USPAT; EPO; JPO; DERWENT; IBM_TDB	2004/08/03 15:29

STIC EIC 2100 Search Request Form

128753

(5)

Today's Date: 2 Aug 2004

What date would you like to use to limit the search?

Priority Date: 1/25/2002 Other:

Name Betit, Jacob F.

AU 2175 Examiner # 80128

Room # 4A30 Phone 305-3735

Serial # 10/058164

Format for Search Results (Circle One):

(PAPER) DISK EMAIL

Where have you searched so far?

(USP) DWPI (EPO) (JPO) ACM IBM TDB

IEEE INSPEC SPI Other PG PUB Google

Is this a "Fast & Focused" Search Request? (Circle One) (YES) NO

A "Fast & Focused" Search is completed in 2-3 hours (maximum). The search must be on a very specific topic and meet certain criteria. The criteria are posted in EIC2100 and on the EIC2100 NPL Web Page at http://ptoweb/patents/stic/stic-tc2100.htm.

What is the topic, novelty, motivation, utility, or other specific details defining the desired focus of this search? Please include the concepts, synonyms, keywords, acronyms, definitions, strategies, and anything else that helps to describe the topic. Please attach a copy of the abstract, background, brief summary, pertinent claims and any citations of relevant art you have found.

using deadlock history to determin. what kind of lock (level) should be put on a data location when it is accessed

Claim 1

lock level can be read/write or read only or can be how much data should be locked

STIC Searcher Geoffrey St.Leger Phone 308-7800

Date picked up 8/2/4 Date Completed 8/3/4

SIRA Search and Information Resources Administration

STIC Search Report
EIC 2100

STIC Database Tracking Number: 128753

TO: Jacob Betit **Location: CPK2, 4A30** **Art Unit : 2175** **Tuesday, August 03, 2004** **Case Serial Number: 10/058164**	**From: Geoffrey St. Leger** **Location: EIC 2100** **PK2-4B30** **Phone: 308-7800** **geoffrey.stleger@uspto.gov**

Search Notes

Dear Examiner Betit,

Attached please find the results of your search request for application 10/058164. I searched Dialog's foreign patent files, technical databases, product announcement files and general files.

Please let me know if you have any questions.

Regards,

Geoffrey St. Leger
4B30/308-7800

```
File 275:Gale Group Computer DB(TM) 1983-2004/Aug 03
         (c) 2004 The Gale Group
File 621:Gale Group New Prod.Annou.(R) 1985-2004/Aug 03
         (c) 2004 The Gale Group
File 636:Gale Group Newsletter DB(TM) 1987-2004/Aug 03
         (c) 2004 The Gale Group
File  16:Gale Group PROMT(R) 1990-2004/Aug 03
         (c) 2004 The Gale Group
File 160:Gale Group PROMT(R) 1972-1989
         (c) 1999 The Gale Group
File 148:Gale Group Trade & Industry DB 1976-2004/Aug 03
         (c)2004 The Gale Group
File 624:McGraw-Hill Publications 1985-2004/Aug 02
         (c) 2004 McGraw-Hill Co. Inc
File  15:ABI/Inform(R) 1971-2004/Aug 02
         (c) 2004 ProQuest Info&Learning
File 647:CMP  Computer Fulltext 1988-2004/Jul W4
         (c) 2004 CMP Media, LLC
File 674:Computer News Fulltext 1989-2004/Jul W4
         (c) 2004 IDG Communications
File 696:DIALOG Telecom. Newsletters 1995-2004/Jul 23
         (c) 2004 The Dialog Corp.
File 369:New Scientist 1994-2004/Jul W4
         (c) 2004 Reed Business Information Ltd.
File 810:Business Wire 1986-1999/Feb 28
         (c) 1999 Business Wire
File 813:PR Newswire 1987-1999/Apr 30
         (c) 1999 PR Newswire Association Inc
File 610:Business Wire 1999-2004/Aug 03
         (c) 2004 Business Wire.
File 613:PR Newswire 1999-2004/Aug 03
         (c) 2004 PR Newswire Association Inc

Set     Items   Description
S1      21525   DEADLOCK??? OR DEAD()LOCK???
S2      16391   (S1 OR LOCK???)(5N)(LEVEL? ? OR TIER? ? OR GRADE? ? OR STR-
             ENGTH OR DEGREE? ? OR KIND? ? OR TYPE? ? OR CATEGOR? OR CLASS-
             ?? OR CLASSIFICATION OR FAMILY OR FAMILIES)
S3       2704   (S1 OR LOCK???)(5N)(HISTORY OR HISTORIES OR LOG? ? OR LOGG-
             ??? OR CACHE? ?)
S4        390   S1(5N)(DATA OR INFORMATION OR NUMBER? ? OR AMOUNT? ? OR ST-
             ATISTIC?? OR PROFILE? ?)
S5         22   S4(5N)(RETAIN??? OR STOR??? OR RECORD??? OR SAV??? OR MAIN-
             TAIN??? OR KEEP??? OR KEPT OR TRACK??? OR MONITOR???)
S6    5422816   DATABASE? ? OR DATA()(BASE? ? OR WAREHOUSE? ?) OR DBMS OR -
             RDBMS OR REPOSITOR??? OR (DATA OR INFORMATION)()MANAGEMENT OR
             FILE? ?
S7          5   S1(50N)S2(50N)(S3 OR S5)(50N)S6
S8          6   S1(50N)S2(50N)(S3 OR S5)
S9         95   S1(50N)S2(50N)S6
S10     10366   (S1 OR LOCK???)(5N)(LEVEL? ? OR TIER? ? OR KIND? ? OR TYPE?
             ?)
S11        85   S1(50N)S10(50N)S6
S12       254   S1(15N)S10
S13        45   S12(50N)S6
S14        33   S1(50N)(S3 OR S5)(50N)S6
S15        78   S7:S8 OR S13:S14
S16        63   RD (unique items)
S17        59   S16 NOT PD>20020125
```

17/3,K/1 (Item 1 from file: 275)
DIALOG(R)File 275:Gale Group Computer DB(TM)

02390138 SUPPLIER NUMBER: 61498430 (USE FORMAT 7 OR 9 FOR FULL TEXT)
Philippines Trial Set For Lotus Chief Vs Microsoft Head 03/24/00. (Company Business and Marketing)
Oliva, Erwin Lemuel G
Newsbytes, NA
March 24, 2000
LANGUAGE: English RECORD TYPE: Fulltext
WORD COUNT: 1312 LINE COUNT: 00106

... been discussing ways of settling the case amicably. At this time, however, both are apparently **deadlocked** over what **type** of conditions will be included in the out-of-court settlement.
Silvino again stressed, "I'm willing to withdraw all the cases **filed** against Lockie as long as there is a fair and mutual respect for our rights...

17/3,K/2 (Item 2 from file: 275)
DIALOG(R)File 275:Gale Group Computer DB(TM)

02176070 SUPPLIER NUMBER: 20527058 (USE FORMAT 7 OR 9 FOR FULL TEXT)
Versant 5.0. (Versant Object Technology's object DBMS) (Server Side) (Software Review) (Column) (Evaluation)
Rennhackkamp, Martin
DBMS, v11, n3, p61(4)
March, 1998
DOCUMENT TYPE: Column Evaluation ISSN: 1041-5173 LANGUAGE: English RECORD TYPE: Fulltext; Abstract
WORD COUNT: 3769 LINE COUNT: 00299

... database are grouped in transactions. An application has to start a session to access any **database** . When an application starts a session, Versant creates various areas in memory, such as an...

...is ended, a new short transaction is automatically started. However, you must have logging and **locking** enabled for the database. If a transaction spans more than one database, a two-phase...

...checked out objects. You switch between multiple long transactions by stopping and restarting sessions.
Short **locks** are set at the object **level** for concurrency control. There are various **types** of short **locks** , such as write locks, update locks, read locks (which can be shared), and so-called...

...used for dirty reads). Applications waiting for locks can be set to time out. Single **database deadlocks** are immediately detected and disallowed by Versant. It uses a timeout mechanism to detect **deadlocks** between multiple **databases** .
Versant has a very extensive set of concurrency control mechanisms - from **types** of **locks** to check-in and check-out operations in the context of multiple long transactions - much more extensive than any relational **DBMS** currently offers. This makes it possible to tune the **database** server's locking behavior very closely to any application's requirements. However, DBAs must know...

17/3,K/3 (Item 3 from file: 275)
DIALOG(R)File 275:Gale Group Computer DB(TM)

01987276 SUPPLIER NUMBER: 18692643 (USE FORMAT 7 OR 9 FOR FULL TEXT)
Performance monitoring. (Server Side) (Column)
Rennhackkamp, Martin

DBMS, v9, n10, p85(4)
Sep, 1996
DOCUMENT TYPE: Column ISSN: 1041-5173 LANGUAGE: English
RECORD TYPE: Fulltext; Abstract
WORD COUNT: 2637 LINE COUNT: 00216

... can query which sessions are holding locks required by other sessions. You can monitor specific **database** metrics and depict the resuits as a graph to understand specific performance trends.
Borland InterBase...

...the InterBase Server Manager and the InterBase Interactive SQL Interface. The InterBase Server Manager displays **database** server statistics and lock manager statistics of the currently connected **database** . The **database** server statistics include transaction ID generation statistics, **database** **file** statistics, and **log** **file** statistics. The **lock** statistics include details on locks requested, granted, and released by the various transactions operating on the **database** . Lock statistics also show **deadlock** scans, **deadlocks** detected, and mutually exclusive **deadlock** waits. All of these statistics are displayed as log **files** in a notepad4ike window.
The InterBase Interactive SQL Interface has various useful settings for performance...

17/3,K/4 (Item 4 from file: 275)
DIALOG(R)File 275:Gale Group Computer DB(TM)
(c) 2004 The Gale Group. All rts. reserv.

01944261 SUPPLIER NUMBER: 18315425 (USE FORMAT 7 OR 9 FOR FULL TEXT)
Desktop DBMSs/application development systems. (1996 Database Buyer's Guide and Client/Server Sourcebook) (Buyers Guide)
DBMS, v9, n6, p57(3)
June 15, 1996
DOCUMENT TYPE: Buyers Guide ISSN: 1041-5173 LANGUAGE: English
RECORD TYPE: Fulltext; Abstract
WORD COUNT: 4123 LINE COUNT: 00351

... 446.
CodeBase 5.1 Sequiter Software Inc., Edmonton, AB, Canada 403-437-2410
Brings multiuser **database** management capabilities and Xbase **file** compatibility to C and C++ programmers. Function names are similar to dBASE dBASE commands. Multiuser C and C++ applications can share data, index, and memo **files** with concurrently running dBASE, FoxPro, and CA-Clipper programs. Data can be shared at the record or **file** **level** over any network. Provides automatic **deadlock** prevention and manual and automatic locking. Allows reading of locked records. Includes a new relate...

17/3,K/5 (Item 5 from file: 275)
DIALOG(R)File 275:Gale Group Computer DB(TM)
(c) 2004 The Gale Group. All rts. reserv.

01944260 SUPPLIER NUMBER: 18315424 (USE FORMAT 7 OR 9 FOR FULL TEXT)
DBA tools. (database administration) (1996 Database Buyer's Guide and Client/Server Sourcebook) (Buyers Guide)
DBMS, v9, n6, p54(3)
June 15, 1996
DOCUMENT TYPE: Buyers Guide ISSN: 1041-5173 LANGUAGE: English
RECORD TYPE: Fulltext; Abstract
WORD COUNT: 4192 LINE COUNT: 00365

... Provides performance data collection, viewing, reporting, analysis, and alerting for DB2/2 and DB2/6000 **databases** . supports more than 200 performance attributes, including buffer pool, lock and **deadlock** , sorting, communication, agent, and **log** -in information. Operates in OS/2 and AIX environments. Reader service $414.
DB2 Visual Explain...

...in a graphical manner, allowing DBAs to use this information to tune SQL statements and **databases** . Provides users with a what-if modeling capability for SQL statements, allowing them to model the impact of various changes in the **database** environment. Operates in OS/2 and AIX environments. Reader service #415.
DBArtisan Embarcadero Technologies Inc...

17/3,K/6 (Item 6 from file: 275)
DIALOG(R)File 275:Gale Group Computer DB(TM)

01944259 SUPPLIER NUMBER: 18315423 (USE FORMAT 7 OR 9 FOR FULL TEXT)
Database servers and host DBMSs.(1996 Database Buyer's Guide and Client/Server Sourcebook)(Buyers Guide)
DBMS, v9, n6, p49(5)
June 15, 1996
DOCUMENT TYPE: Buyers Guide ISSN: 1041-5173 LANGUAGE: English
RECORD TYPE: Fulltext; Abstract
WORD COUNT: 5960 LINE COUNT: 00519

... at the low level, ISAM level, or ODBC level. Page locks are never performed; read **locks** are not required; record- **level** looking is available with automatic short-term key- **level** **locking** . Includes extensive anti- **deadlock** detection/resolution, caching/index compression to reduce I/O; full OLTP with complete roll-forward and roll-back; intermittent save points; **file** mirroring; multiple contiguous batch and set operations; and full administrative API. Supports DOS Extended, Windows ...

17/3,K/7 (Item 7 from file: 275)
DIALOG(R)File 275:Gale Group Computer DB(TM)

01941532 SUPPLIER NUMBER: 18303694 (USE FORMAT 7 OR 9 FOR FULL TEXT)
Performance monitoring in a client/server environment. (Technology Information)
Kuay, Daniel D.; Kenney, Paul
Enterprise Systems Journal, v11, n5, p54(7)
May, 1996
ISSN: 1053-6566 LANGUAGE: English RECORD TYPE: Fulltext; Abstract
WORD COUNT: 3415 LINE COUNT: 00280

... the different types of buffer cache, such as data, library, dictionary and procedure? If the **database** system is storage-constrained, are transactions being locked out? Limits reached on the maximum number...

...holding locks longer than needed or holding too many locks can also cause timeouts and **deadlocks** .
Investigate other **types** of resource contention across executing applications and sessions within the **RDBMS** . Slower access to internal server resources by executing applications frequently causes systemwide performance problems, including...

17/3,K/8 (Item 8 from file: 275)
DIALOG(R)File 275:Gale Group Computer DB(TM)

01922772 SUPPLIER NUMBER: 18129547 (USE FORMAT 7 OR 9 FOR FULL TEXT)
Making the right connection. (access to remote data) (Technology Information)
Saracco, C.M.
Enterprise Systems Journal, v11, n3, p22(5)
March, 1996
ISSN: 1053-6566 LANGUAGE: English RECORD TYPE: Fulltext; Abstract

WORD COUNT: 2653 LINE COUNT: 00226

... The inherent complexity of such an environment as well as functional limitations of early distributed **database** offerings are among the reasons for this.

Distributed **database** environments consist of multiple independent DBMSes interconnected by a network. These DBMSes are capable of...

...is physically distributed and managed by multiple DBMSes is a common goal of a distributed **database** environment. Such a system is often said to support location transparency.

Developing support for such a distributed **database** environment is no easy task, especially since heterogeneity is often a requirement. Integrity issues must be addressed, which can impact **lock** management, **log** management and other **DBMS** functions. Performance issues may prompt the need for global optimization, enhancements to system catalogs, a means to cope with global **deadlocks** and **database** design alternatives such as horizontal or vertical partitioning. In addition, such "simple" issues as data...

...this environment.

For these reasons and others, vendors offer varying degrees of support for distributed **database** technology today. Figure 1 categorizes these levels of support, with the most primitive level at...

17/3,K/9 (Item 9 from file: 275)
DIALOG(R)File 275:Gale Group Computer DB(TM)
(c) 2004 The Gale Group. All rts. reserv.

01908962 SUPPLIER NUMBER: 18046547 (USE FORMAT 7 OR 9 FOR FULL TEXT)
Parallel processing with DB2 PE. (IBM's DBMS) (Server Side) (Product Information)(Column)
Miller, Stewart
DBMS, v9, n3, p75(4)
March, 1996
DOCUMENT TYPE: Column ISSN: 1041-5173 LANGUAGE: English
RECORD TYPE: Fulltext; Abstract
WORD COUNT: 3985 LINE COUNT: 00326

... the same node. DB2 PE instructs each processor to access only the portion of the **database** that it owns locally; therefore, a processor does not have to request access permission from...

...approach eliminates the need for a global lock table.

DB2 PE is built upon the **Data Management** Services layer of the nonparallel DB2/6000 engine, but IBM has completely reworked the communication services. The Data Protection Services (DPS) layer of DB2/6000 (responsible for **locking** , **logging** , and recovery) can activate more than one process and involve more than one node at one time. DPS extensions use a control message interface for global **deadlock** detection, two-phase commits, and recovery from system failures.

I/O shipping is an alternative...

17/3,K/10 (Item 10 from file: 275)
DIALOG(R)File 275:Gale Group Computer DB(TM)
(c) 2004 The Gale Group. All rts. reserv.

01820660 SUPPLIER NUMBER: 17385088 (USE FORMAT 7 OR 9 FOR FULL TEXT)
ISAM Manager database library. (from Nildram Software) (Shareware Spotlight) (Software Review)(Evaluation)
Volkman, Victor R.
Windows-DOS Developer's Journal, v6, n7, p47(7)
July, 1995
DOCUMENT TYPE: Evaluation ISSN: 1059-2407 LANGUAGE: English
RECORD TYPE: Fulltext; Abstract
WORD COUNT: 2946 LINE COUNT: 00244

... the introduction, you can license ISAM Manager in single-user (standalone) or multi-user (network **file** sharing) configurations. It is relatively easy to code a program that will work in either mode with a simple recompilation. The main considerations in implementation relate to **cache** and **deadlock** avoidance.

In the standalone version, you can set the number of cache buffers to tune...

...tell each other to flush caches after a write occurs. Since this is a peer **database** rather than client/server, there is no server to provide caching either. Of course, most...
...servers, such as Novell, allow considerable flexibility independent of applications.

According to Ullman 19821, a **deadlock** is a situation in which each member of a set S of two or more...

...to the two-phase commit).

The two-phase protocol is enforced within a given data **file** by making any function which retrieves a record implicitly cause a lock to occur. The...

17/3,K/11 (Item 11 from file: 275)
DIALOG(R)File 275:Gale Group Computer DB(TM)

01817079 SUPPLIER NUMBER: 17369260 (USE FORMAT 7 OR 9 FOR FULL TEXT)
DB2 Version 4 for performance.(IBM's DB2 V4 database management system)
Kitay, Daniel D.
Enterprise Systems Journal, v10, n6, p42(6)
June, 1995
ISSN: 1053-6566 LANGUAGE: English RECORD TYPE: Fulltext; Abstract
WORD COUNT: 2782 LINE COUNT: 00223

... lock structure and from one to 60 GBPs.

The SCA saves group-wide status and **database** exception information. The lock structure stores global lock information needed to manage cross-system concurrency...

...in addition to each local DB2 member's Virtual Pool (VP) and Hiper Pool (HP) **caches** .

DS introduces many new global **locking** optimization techniques to ensure data integrity while minimizing the amount of cross-system communication needed...

...is up-dating, there is extra "global" processing for shared tablespace, table, page or row- **level** **lock** requests. Cross-system **lock** suspensions, timeouts and **deadlocks** add another **level** of complexity in monitoring and resolving contention problems.

Other critical factors for cross-system availability...

17/3,K/12 (Item 12 from file: 275)
DIALOG(R)File 275:Gale Group Computer DB(TM)

01804827 SUPPLIER NUMBER: 17155740 (USE FORMAT 7 OR 9 FOR FULL TEXT)
Tools and utilities.(1995 Database Buyer's Guide and client/server sourcebook)(Buyers Guide)
DBMS, v8, n6, p72(29)
May 15, 1995
DOCUMENT TYPE: Buyers Guide ISSN: 1041-5173 LANGUAGE: English
RECORD TYPE: Fulltext; Abstract
WORD COUNT: 45154 LINE COUNT: 03869

... Provides performance data collection, viewing, reporting, analysis, and alerting for DB2/2 and DB2/6000 **databases** . Supports more than 200

performance attributes, including buffer pool, lock and **deadlock** , sorting, communication, agent, and **log** -in information. Operates in OS/2 and AIX environments. Reader service #704.

DB2 Visual Explain...

...in a graphical manner, allowing DBAs to use this information to tune SQL statements and **databases** . Provides users with a what-if modeling capability for SQL statements, allowing them to model the impact of various changes in the **database** environment. Operates in OS/2 and AIX environments. Reader service #705.

DBA Lynx

Business Planning...SQL translators and a menu-driven system. Additional tools let users create, drop, and resize **databases** ; backup, restore, and view space usage; create or drop tables and modify structure while preserving...

...set default languages, and grant and revoke permissions. $695. Reader service #742.

SQL Index Inspector & **DeadLock** Predictor

The Development Group for Advanced Tech. Inc., Upper Saddle River, NJ

201-825-9511

Identifies SQL statements not covered by indexes and predicts the occurrence of positional-based **deadlocks** in **database** procedures and triggers. The **database** transactions and SQL statements involved in a predicted **deadlock** scenario are presented in the SQL **Deadlock** Browser. Dependency Graphs of the transactions are displayed in Graphical Browsers. Positional-based **deadlocks** in SQL code may be resolved prior to application deployment. **Deadlock** prediction may be performed on a batch or interactive basis. The SQL Inspector and **Deadlock** Predictor **save data** administrators and developers days of research and analysis in the troubleshooting of **deadlock** conditions. Improves application service and determines potential **deadlocks** prior to application deployment. Available for Microsoft SQL Server, Oracle, and Sybase. Runs on Sun Solaris, Windows 3.1, and Windows NT, and connects to any network **database** server. $899. Reader service #743.

SQL Monitor

Sybase Inc., Emeryville, CA

510-596-3500

A...

17/3,K/13 (Item 13 from file: 275)

01804823 SUPPLIER NUMBER: 17155732 (USE FORMAT 7 OR 9 FOR FULL TEXT)

Microcomputer DBMSs and application development systems. (1995 Database Buyer's Guide and client/server sourcebook) (Buyers Guide)

DBMS, v8, n6, p52(5)
May 15, 1995
DOCUMENT TYPE: Buyers Guide ISSN: 1041-5173 LANGUAGE: English
RECORD TYPE: Fulltext; Abstract
WORD COUNT: 6434 LINE COUNT: 00552

... 687.

CodeBase 5.1

Sequiter Software Inc., Edmonton, Alberta, CANADA

403-437-2410

Brings multiuser **database** management capabilities and Xbase **file** compatibility to C and C++ programmers. Function names are similar to dBASE commands. Multiuser C...

...running dBASE, FoxPro, and CA-Clipper programs. Data can be shared at the record or **file level** over any network. Provides automatic **deadlock** prevention and manual and automatic locking. Allows reading of locked records. Includes a new relate...

17/3,K/14 (Item 14 from file: 275)
DIALOG(R)File 275:Gale Group Computer DB(TM)

01702420 SUPPLIER NUMBER: 16250298 (USE FORMAT 7 OR 9 FOR FULL TEXT)
SQL databases: the great leap forward. (includes related articles on Editors' Choices, suitability to task ratings, performance tests, Intel-based symmetric multiprocessing, competing with RISC systems, price/performance ratios, summary of features) (Software Review) (overview of six evaluations of SQL databases) (individual evaluation records searchable under "SQL Databases The Great Leap Forward") (Evaluation)
Butler, Brian; Mace, Thomas
PC Magazine, v13, n17, p241(17)
Oct 11, 1994
DOCUMENT TYPE: Evaluation ISSN: 0888-8507 LANGUAGE: ENGLISH
RECORD TYPE: FULLTEXT; ABSTRACT
WORD COUNT: 9435 LINE COUNT: 00738

... on this extremely demanding test was Oracle7. Its high score is attributable to its record- **locking** scheme and efficient **log** management, features that have been part of the product for quite some time. The engine ...

...Oracle7 can write in a group commit is limited only by the operating system. Record- **level** **locking** provided optimal concurrency: No **deadlocks** occurred during the execution of the test. Oracle7 did not use its Discrete Transaction feature...

...how much work Microsoft has done to speed up the transaction-processing aspect of the **database** 's engine. One new feature, asynchronous checkpoints, allows translation processing to continue during the checkpoint process. A Lazy Writer feature, also new to this release, lets the **database** engine clean up dirty **database** pages in the background, minimizing the work required during the checkpoint process. Microsoft SQL Server...

17/3,K/15 (Item 15 from file: 275)
DIALOG(R)File 275:Gale Group Computer DB(TM)

01688545 SUPPLIER NUMBER: 15356060 (USE FORMAT 7 OR 9 FOR FULL TEXT)
Tools and utilities. (1994 Database Buyer's Guide and Client/Server Sourcebook) (Buyers Guide)
DBMS, v7, n6, p63(29)
June 15, 1994
DOCUMENT TYPE: Buyers Guide ISSN: 1041-5173 LANGUAGE: ENGLISH
RECORD TYPE: FULLTEXT; ABSTRACT
WORD COUNT: 46074 LINE COUNT: 03903

... object code consisting of 210 procedures and functions, it supports any data type, including text **files** , PICT images, sound, and movies, and can be used with most C, C++, and Pascal...

...groups of records; multiple databases open within one file; easy data reorganization; three recovery levels; **deadlock** detection; messaging system between clients; runtime debugging; Data Browser; and modified b-tree indexing system...Lets users create data files, modify existing ones, or convert Notron Guides with the Data **File** Maker. $70-$800.
LIBRARIES
ASE 2.6 Black & and White International Inc., New York, NY...
...set of functions that provides easy access to and modification of dBASE and CA-Clipper **files** . Supports both DOS and Windows; network-ready. $69.95.
Btrv++ Classic software Inc., Ann Arbor, MI 313-677-0732
A C++ library of 15 major classes providing a C/C++ **database** interface to Btrieve. Implements all Btrieve operations as member

functions, as well as many value-added functions for SQL queries, **file** and index creation, iteration, extended operations, and global Btrieve management. Record- and field-based access...

...significant OLTP applications with pretested interface and functional logic. These applications will automatically utilize every **DBMS** supported by SQLWindows, including Oracle, DB2, SQL Server, HP-Allbase, SQLBase, Informix, and AS/400...

...support are also available.
C-Index/II Trio Systems, Pasadena, CA 818-584-9706
A **data - management** library for C suited for development of high-quality products and turnkey applications. Offers B...and up.
CodeBase 5.1 Sequiter Software Inc., Newmarket, NH 403-437-2410
Brings multiuser **database** -management capabilities and Xbase **file** compatibility to C and C++ programmers. CodeBase functions are named similar to familiar dBASE commands, minimizing the learning curve. Multiuser C and C++ applications can share data, index, and memo **files** with concurrently running dBASE, FoxPro, and CA-Clipper programs. Data can be shared at the record or **file levels** over any network. Provides automatic **deadlock** prevention and manual and automatic locking. It also allows reading of locked records. Includes a...

17/3,K/16 (Item 16 from file: 275)
DIALOG(R)File 275:Gale Group Computer DB(TM)

01636055 SUPPLIER NUMBER: 14822467 (USE FORMAT 7 OR 9 FOR FULL TEXT)
Network programming basics. (CA-Clipper database application development software) (Clipper Basics) (Column) (Tutorial)
Gutierrez, Dan D.
Data Based Advisor, v11, n12, p145(5)
Dec, 1993
DOCUMENT TYPE: Tutorial ISSN: 0740-5200 LANGUAGE: ENGLISH
RECORD TYPE: FULLTEXT; ABSTRACT
WORD COUNT: 3158 LINE COUNT: 00262

... no time out provisions in the locking routine).
The solution is to consistently open the **files** in the same order. That way, resource conflicts are settled immediately, before a deadlock can ...

...prevention.
Note that since you may now lock more than one record in a single **database** via the **Lock** List, the potential for record- **level deadlock** exists, too.
Testing multi-user apps
As a beginning Clipper developer, you may be concerned...

17/3,K/17 (Item 17 from file: 275)
DIALOG(R)File 275:Gale Group Computer DB(TM)

01625966 SUPPLIER NUMBER: 14485610 (USE FORMAT 7 OR 9 FOR FULL TEXT)
IT boom. (information technology; Foster Yeoman Ltd. uses Oracle Corp.'s Oracle 7 relational database management system)
Dudman, Jane
DEC User, p17(2)
Sept, 1993
ISSN: 0263-6530 LANGUAGE: ENGLISH RECORD TYPE: FULLTEXT; ABSTRACT
WORD COUNT: 1753 LINE COUNT: 00130

...ABSTRACT: use. Other benefits from the system include its use of roll-back segments to store **database** changes and its row **level locking** which allows access to more concurrent users without risk of a **deadlock**. The company has also been impressed by the performance improvements found

in each new version...

17/3,K/18 (Item 18 from file: 275)
DIALOG(R)File 275:Gale Group Computer DB(TM)

01619141 SUPPLIER NUMBER: 14413848 (USE FORMAT 7 OR 9 FOR FULL TEXT)
Minigrams extra.
Computergram International, CGI09100025
Sept 10, 1993
ISSN: 0268-716X LANGUAGE: ENGLISH RECORD TYPE: FULLTEXT
WORD COUNT: 519 LINE COUNT: 00042

... John Schoof, ousted earlier this year as chief executive of Artisoft Inc, Tucson, Arizona has **filed** a demand for arbitration and claims $1.5m in salary and bonus and vesting of...

...the Artisoft board, two to replace vacancies, and a third to prevent the chance of **deadlock** because the board currently **numbers** six people; Schoof has **retained** his 21% stake in the firm, giving him the right to make the nominations.
- o...

17/3,K/19 (Item 19 from file: 275)
DIALOG(R)File 275:Gale Group Computer DB(TM)

01616362 SUPPLIER NUMBER: 13906049 (USE FORMAT 7 OR 9 FOR FULL TEXT)
Process industry. (boosting processor power on a DEC VAX minicomputer: last in a series of articles on VAX tuning) (Tutorial)
Sethi, Joginder
DEC User, p31(2)
April, 1993
DOCUMENT TYPE: Tutorial ISSN: 0263-6530 LANGUAGE: ENGLISH
RECORD TYPE: FULLTEXT; ABSTRACT
WORD COUNT: 1456 LINE COUNT: 00112

...ABSTRACT: such costs. The described techniques address such factors as running batch jobs, page fault rate, **caches** , **locking** activity, optimum bucket sizes use of terminal servers for load balancing, prioritizing jobs, use of access control lists, DCL procedures, direct memory access, **deadlock** searches, use of backup utilities and reduction of **file** fragmentation.

17/3,K/20 (Item 20 from file: 275)
DIALOG(R)File 275:Gale Group Computer DB(TM)

01613921 SUPPLIER NUMBER: 13901763 (USE FORMAT 7 OR 9 FOR FULL TEXT)
Tools and utilities. (software packages that help database developers prototype and design applications, query, and create help systems, among other uses) (1993 Database Buyer's Guide Special Issue) (Buyers Guide)
DBMS, v6, n7, p63(33)
June 15, 1993
DOCUMENT TYPE: Buyers Guide ISSN: 1041-5173 LANGUAGE: ENGLISH
RECORD TYPE: FULLTEXT; ABSTRACT
WORD COUNT: 45702 LINE COUNT: 03876

... code consisting of 210 implemented procedures and functions. It supports any data type, including text **files** , PICT images, sound, and movies, and can be used with most C, C++, and Pascal environments. Features include: sets, for high-speed retrieval of groups of records; multiple **databases** open within one **file** ; easy on-the-fly data reorganization; three recovery **levels** ; **deadlock** detection; messaging system between clients; run-time debugging; Data Browser; and modified B-tree indexing...

...NY 718-793-7963, 800-535-3267
An ANSI standards-based single- and multiuser SQL **database** engine for C and C++ developers. It supports ODBC and is available on DOS, Windows ...

17/3,K/21 (Item 21 from file: 275)
DIALOG(R)File 275:Gale Group Computer DB(TM)

01551703 SUPPLIER NUMBER: 13074933 (USE FORMAT 7 OR 9 FOR FULL TEXT)
Locking and referential integrity in Oracle. (Oracle 6.0 data base management system) (Server Specific) (Tutorial)
Fratarcangeli, Claudio
DBMS, v5, n13, p81(6)
Dec, 1992
DOCUMENT TYPE: Tutorial ISSN: 1041-5173 LANGUAGE: ENGLISH
RECORD TYPE: FULLTEXT; ABSTRACT
WORD COUNT: 5629 LINE COUNT: 00471

... development editions), it is still uncertain whether the locking scheme used by the kernel avoids **deadlock** , or whether it obtains the weakest **lock** possible at the lowest **level** of granularity in order to maximize concurrency. This could have serious implications. The scheme the ...

...internally may conflict with a global deadlock-avoidance locking scheme used in an application. The **database** administrator may need to implement a deadlock-avoidance locking scheme consistent with the V7 kernel...

17/3,K/22 (Item 22 from file: 275)
DIALOG(R)File 275:Gale Group Computer DB(TM)

01445404 SUPPLIER NUMBER: 11178881 (USE FORMAT 7 OR 9 FOR FULL TEXT)
Automated and repeatable randomness. (requirements for a successful benchmark test)
DBMS, v4, n9, p56(1)
August, 1991
ISSN: 1041-5173 LANGUAGE: ENGLISH RECORD TYPE: FULLTEXT; ABSTRACT
WORD COUNT: 854 LINE COUNT: 00063

... data-load portion of the benchmark, it is essential that every transaction succeed if the **DBMS** products being compared are to face the same number of rows in the test **data** . **base** . To that end, the benchmark control logic includes logic that retries a transaction in the event of a temporarily lost network connection or a rollback due to record- or page-**level** **deadlocks** .
All **DBMS** -specific control logic - for example, a retry in the event of a network failure for an Oracle client involves dropping all open cursors and recork necting to the **database** after a brief random interval - is pulled into a separate library to ease the task...

17/3,K/23 (Item 23 from file: 275)
DIALOG(R)File 275:Gale Group Computer DB(TM)

01443837 SUPPLIER NUMBER: 11139657 (USE FORMAT 7 OR 9 FOR FULL TEXT)
Oracle puts the real thing on Mac. (Oracle Server database for the Macintosh) (includes related article on features) (Software Review) (evaluation)
Keenan, Vernon
MacWEEK, v5, n27, p112(1)
August 6, 1991
DOCUMENT TYPE: evaluation ISSN: 0892-8118 LANGUAGE: ENGLISH

RECORD TYPE: FULLTEXT; ABSTRACT
WORD COUNT: 1450 LINE COUNT: 00116

... Feb. 19).
Oracle Server for Macintosh is a complete port of Oracle's Version 6 **RDBMS** server, which features on-line transaction-processing-oriented enhancements. This server supports user authentication, row- **level** security, relational record **locking** , **deadlock** resolution, PL/SQL (a new programming language) and transaction rollback. Oracle Server for Macintosh can...

17/3,K/24 (Item 24 from file: 275)
DIALOG(R)File 275:Gale Group Computer DB(TM)

01442705 SUPPLIER NUMBER: 11007077 (USE FORMAT 7 OR 9 FOR FULL TEXT)
The database shuffle: third-party 4GLs and database tool sets put on their platform shoes and hustle off to other architectures. (fourth-generation languages)
McLachlan, Gordon
HP Professional, v5, n7, p56(4)
July, 1991
ISSN: 0896-145X LANGUAGE: ENGLISH RECORD TYPE: FULLTEXT; ABSTRACT
WORD COUNT: 2095 LINE COUNT: 00165

... client-server overhead for complex actions and allowing user-defined functions.
* Transaction managers to synchronize **database** updates and back out incomplete transactions.
* Roll-back and recovery features.
* Page or row- **level** **locks** .
* Transaction **deadlock** detection.
ISO also has taken up the SQL challenge and is working on a Remote Data Access (RDA) specification for linking remote **databases** over networks. ANSI is now also finalizing the specifications for SQL Level 2, that will...

17/3,K/25 (Item 25 from file: 275)
DIALOG(R)File 275:Gale Group Computer DB(TM)

01437425 SUPPLIER NUMBER: 10871126 (USE FORMAT 7 OR 9 FOR FULL TEXT)
The road to client-server Nirvana: all signs say SQL servers and RDBMSs are the way to go. (Databases)
McLachlan, Gordon
LAN Computing, v2, n8, p17(2)
April 9, 1991
ISSN: 1055-1808 LANGUAGE: ENGLISH RECORD TYPE: FULLTEXT; ABSTRACT
WORD COUNT: 1591 LINE COUNT: 00121

... client-server overhead for complex actions and allowing user-defined functions,
* Transaction managers to synchronize **database** updates and back out incomplete transactions,
* Roll-back and recovery features,
* Page or row- **level** **locking** of data, and
* Transaction **deadlock** detection.
In addition, not even a standard SQL will provide interoperability between RDBMSs. True interoperability...

...to define the data formats and communications protocols between the SQL APIs and the underlying **database** manager. This is where

17/3,K/26 (Item 26 from file: 275)
DIALOG(R)File 275:Gale Group Computer DB(TM)

01430512 SUPPLIER NUMBER: 10343632 (USE FORMAT 7 OR 9 FOR FULL TEXT)
Wearing the same Tuxedo. (AT and T Unix System Laboratories Tuxedo System/T transaction processing monitor)
Jones, Keith
Computer Weekly, n1246, p26(2)
Jan 24, 1991
ISSN: 0010-4787 LANGUAGE: ENGLISH RECORD TYPE: FULLTEXT; ABSTRACT
WORD COUNT: 1530 LINE COUNT: 00124

... volume TP network and supporting two phase commit across a distributed TP network. All the **database** tables affected by a global transaction are checked before committing an update.

The local resource manager, the term for the **rdbms** in TP parlance, handles transaction commit and rollback at each site. But Tuxedo/T co-ordinates recovery in the event of site failure, network failure or global **deadlock** .

To comminicate with different **types** of resource manager Tuxedo/T employs the XA interface defined by the XTP committee of...

17/3,K/27 (Item 27 from file: 275)
DIALOG(R)File 275:Gale Group Computer DB(TM)
(c) 2004 The Gale Group. All rts. reserv.

01425134 SUPPLIER NUMBER: 10567388 (USE FORMAT 7 OR 9 FOR FULL TEXT)
Oracle Server: easy SQL for Mac nets. (Oracle Corp.'s Macintosh releases of Oracle Server and Oracle Card) (product announcement)
Keenan, Vernon
MacWEEK, v5, n13, p1(2)
April 2, 1991
DOCUMENT TYPE: product announcement ISSN: 0892-8118 LANGUAGE: ENGLISH RECORD TYPE: FULLTEXT; ABSTRACT
WORD COUNT: 739 LINE COUNT: 00061

...ABSTRACT: Odesta Corp's Double Helix and Blyth Software Inc's Omnis for high-end Macintosh **database** development. The program includes many mainframe on-line transaction processing features, such as user authentication, row- **level** security, relational record **locking** and **deadlock** resolution. The design and use of the program on the Macintosh is described, as is...

... Odesta Corp.'s Double Helix and Blyth Software Inc.'s Omnis for developing high-end **database** systems on the Mac.

The server features several mainframe-standard features for on-line transaction processing, including user authentication, row- **level** security, relational record **locking** , **deadlock** resolution and transaction rollback processing.

The only networked clients Oracle Server for Macintosh will support ...

17/3,K/28 (Item 28 from file: 275)
DIALOG(R)File 275:Gale Group Computer DB(TM)
(c) 2004 The Gale Group. All rts. reserv.

01367990 SUPPLIER NUMBER: 08714902 (USE FORMAT 7 OR 9 FOR FULL TEXT)
Coping with multiuser contention. (structured query language functions) (tutorial)
Sayles, Jonathan S.
Data Based Advisor, v8, n6, p41(2)
June, 1990
DOCUMENT TYPE: tutorial ISSN: 0740-5200 LANGUAGE: ENGLISH
RECORD TYPE: FULLTEXT; ABSTRACT
WORD COUNT: 1027 LINE COUNT: 00077

... 2 and wants row 1. Contention is bad enough (system response degrades, sometimes to intolerable **levels**), but **deadlock** is even worse.

A typical system response to **deadlock** is for the **DBMS** to choose one user as a victim; then it rolls that user out of the **database** (while canceling their access requests) with a rude error message. Deadlock and contention can occur on any multiuser **database** system. They occur depending on:

* What the concurrent users are trying to do. Are they...

17/3,K/29 (Item 29 from file: 275)
DIALOG(R)File 275:Gale Group Computer DB(TM)

01363153 SUPPLIER NUMBER: 08528222 (USE FORMAT 7 OR 9 FOR FULL TEXT)
Measuring UNIX database performance: vendors, standards groups address lack of tools.
Winston, Alan
Software Magazine, v10, n6, p78(2)
May, 1990
ISSN: 0897-8085 LANGUAGE: ENGLISH RECORD TYPE: FULLTEXT; ABSTRACT
WORD COUNT: 1511 LINE COUNT: 00122

... each drive. How many cache hits do you get?"
The information available about a Unify **database** today covers the locks and the logical and physical I/Os, and that comes basically...

...available from a performance monitor to be included in the next release of the Unify **database** package, according to Osberg.
ONLINE MONITOR IN DEVELOPMENT
Gilbert Wai, director of product marketing for Informix Software, Menlo Park, Calif., commented on the Informix OnLine **DBMS** Monitor now in development. "A menu-driven user interface allows the **database** administrator to tell what's going on in the **database** with respect to performance. You can spot bottlenecks or make appropriate changes when your application...

...said.
"You can see how you're doing with cache hit ratios," he continued. (The **cache** stores system tables, **locks** and **logs** before they are flushed to disk. If they can be retrieved from cache rather than...

...out of the buffer, how many disk writes, how many are buffered, how much is **cached** how many **deadlocks** are detected," Wai said. "Primarily, this gives you a sense of what's going on...

...locking granularity?" (That is, are you locking records, pages or whole tables?) If you can **lock** at row-level instead of table-level, other applications don't have to wait to...

...distributed queries. We describe all the things that are used for the query processing: full- **file** read vs. indexed read, for example. It explains to the programmer what course the optimizer...

17/3,K/30 (Item 30 from file: 275)
DIALOG(R)File 275:Gale Group Computer DB(TM)

01345241 SUPPLIER NUMBER: 08029718 (USE FORMAT 7 OR 9 FOR FULL TEXT)
B-Tree Filer v. 5.0. (Software Review) (utility for Turbo Pascal) (evaluation)
Powell, Jim
Computer Language, v7, n1, p99(1)
Jan, 1990
DOCUMENT TYPE: evaluation ISSN: 0749-2839 LANGUAGE: ENGLISH
RECORD TYPE: FULLTEXT; ABSTRACT
WORD COUNT: 844 LINE COUNT: 00066

... and the documentation provides several ways to ensure this. The network version also supports three **levels** of **locking** : all **file** blocks (**deadlock** is prevented at this **level**), single **file** block, and single record. You can add other locking code (for shareable locking under Novell...

...Functions are written so that moving from single-user to multiuser functions is simple.

Several **database** maintenance utilities are offered. REBUILD is used in the event that the **file** block was not closed properly (for instance, to recover from a power failure) and must...

17/3,K/31 (Item 31 from file: 275)
DIALOG(R)File 275:Gale Group Computer DB(TM)

01342649 SUPPLIER NUMBER: 08485240
OLTP migrates to PC LANs. (On-line transaction processing)
Letson, Russell
Systems Integration, v23, n5, p40(6)
May, 1990
ISSN: 1044-4262 LANGUAGE: ENGLISH RECORD TYPE: ABSTRACT

ABSTRACT: On-line transaction processing (OLTP) involving several users accessing and altering a **database** has traditionally been handled by computer systems featuring minicomputer or mainframe hardware, but technological advances...

...can achieve the concurrency and data-integrity necessary for OLTP with such features as record **locking** , transaction **logging** , crash recovery and **deadlock** control. Companies such as Information Builders Inc, Informix Software Inc, Progress Software Corp, Data Access...

17/3,K/32 (Item 32 from file: 275)
DIALOG(R)File 275:Gale Group Computer DB(TM)

01329193 SUPPLIER NUMBER: 09492897
Database transactions. (ODBMS) (column)
Loomis, Mary E.S.
Journal of Object-Oriented Programming, v3, n3, p54(5)
Sept-Oct, 1990
DOCUMENT TYPE: column ISSN: 0896-8438 LANGUAGE: ENGLISH
RECORD TYPE: ABSTRACT

ABSTRACT: An object **data base** supports several concurrent users; multiple programs can be accessing objects in the **data base** simultaneously. The **data base** management system (**DBMS**) must prevent these users and programs from interfering with each other; this is sometimes called...

...DBMSs use Transaction Management to meet the concurrency control challenge. Transaction Management must maintain the **data base** consistency, even if the system crashes and no matter how many users or programs are accessing the objects. There are a number of ways to do Transaction Management: **logging** , roll back, roll forward, **locking** , **deadlock** avoidance, and **deadlock** detection. There are three basic properties to a transaction: it is application-defined, obeying the...
...does not happen; and it cannot be undone. Programmers must have some way in the **DBMS** of specifying where a transaction begins and ends. The commercial **DBMS** environment takes Transaction Management for granted.

17/3,K/33 (Item 33 from file: 275)
DIALOG(R)File 275:Gale Group Computer DB(TM)

01319548 SUPPLIER NUMBER: 07514332 (USE FORMAT 7 OR 9 FOR FULL TEXT)
Rites of passage. (database interoperability)
McGuckin, Paul
UNIX Review, v7, n8, p48(6)
August, 1989
ISSN: 0742-3136 LANGUAGE: ENGLISH RECORD TYPE: FULLTEXT; ABSTRACT
WORD COUNT: 2892 LINE COUNT: 00236

... model expected by an application program, problems can arise. If an application program expecting row- **level locking** is run against a table-locking **DBMS** , for example, **deadlocks** may occur at runtime. Similarly, if an application program that expects a multiversion read-consistency model executes an hourly subtotal query against a page-locking **DBMS** , the program might receive incorrect results. Vendor differences over concurrency-control strategies are exacerbated by...

17/3,K/34 (Item 34 from file: 275)
DIALOG(R)File 275:Gale Group Computer DB(TM)

01293114 SUPPLIER NUMBER: 07086130 (USE FORMAT 7 OR 9 FOR FULL TEXT)
A database server odyssey. (database management)
Finkelstein, Richard
PC Tech Journal, v7, n3, p42(13)
March, 1989
DOCUMENT TYPE: evaluation ISSN: 0738-0194 LANGUAGE: ENGLISH
RECORD TYPE: FULLTEXT; ABSTRACT
WORD COUNT: 6827 LINE COUNT: 00568

... GRANT statements for resource and table privileges. XDB-Server has GRANT commands on tables and **databases** , while SQL Server and SQLBase use GRANT commands on tables and resources. THE TIP OF PERFORMANCE Factors that affect **database** performance include the quality of the locking manager (does it use intent **locks** , how does it handle **deadlocks**), **levels** of **locking** strategies (table **locks** versus page locks), SQL optimization, operating-system overhead (single process/multithreaded versus multiprocess), and network...

17/3,K/35 (Item 35 from file: 275)
DIALOG(R)File 275:Gale Group Computer DB(TM)

01238962 SUPPLIER NUMBER: 06199936 (USE FORMAT 7 OR 9 FOR FULL TEXT)
The multiuser perspective. (Software Review) (three multiuser database management systems are evaluated: Ashton-Tate's dBASE III Plus 1.1 , Borland International's Paradox 2.0, and DataEase International's DataEase LAN 1.02) (evaluation)
Browning, Dave
PC Tech Journal, v6, n2, p114(12)
Feb, 1988
DOCUMENT TYPE: evaluation ISSN: 0738-0194 LANGUAGE: ENGLISH
RECORD TYPE: FULLTEXT; ABSTRACT
WORD COUNT: 7557 LINE COUNT: 00599

... in the data manager's programming language? Are duplicate author records added to the Author **file** ? Does the second user detect the new Author record added by the first user before...

...to the second user's transaction? Can the second Article record end up in the **database** without the corresponding Author **record** ?

Does the **data** manager detect **deadlock** if two or more transactions have locked some resources and are waiting to lock other resources already locked by another transaction? If the data manager detects **deadlock** , does it resolve the conflict and how?

If the data manager provides transaction processing, does...

...transaction processing?
Security. Data security is one of the most pressing problems in the shared **database** environment today. The criteria for evaluating multiuser data managers should include answers to these questions...vast assortment of possible combinations for data sharing in both interactive and programmed mode. Effective **database** administration, however, requires a comprehensive knowledge of DataEase processing methodology and a thorough understanding of data-sharing concepts.
All three data managers provide data security via passwords and access **levels** . None detects nor recovers from **deadlock** . Paradox's ability to lock sets of resources and DataEase LAN's variety of nonprocedural...

17/3,K/36 (Item 36 from file: 275)
DIALOG(R)File 275:Gale Group Computer DB(TM)

01210281 SUPPLIER NUMBER: 05117137 (USE FORMAT 7 OR 9 FOR FULL TEXT)
PC data jumps to multiuser systems. (multiuser microcomputer data base management systems)
Karon, Paul
PC Week, v4, n33, p106(1)
Aug 18, 1987
ISSN: 0740-1604 LANGUAGE: ENGLISH RECORD TYPE: FULLTEXT; ABSTRACT
WORD COUNT: 1337 LINE COUNT: 00107

ABSTRACT: Users planning a multiuser microcomputer **data base** management system must change the way they perceive the data on the system as well...

...themselves with new hardware and software. Incorrect data or inadvertently erased data in a multiuser **DBMS** can be trouble for an entire group, which makes the integrity of the data more important. **Data base** administrators in a minicomputer or mainframe environment are well aware of the importance of data...

...systems may not be. Factors departmental computer system managers must consider when installing a multiuser **DBMS** include record- **level lockout** , a **deadlock** facility, and **data base** recovery and backup.
... their purposes, or whether the information must be the absolute latest available.
Two non-SQL **database** managers, DataFlex from Data Access Corp., of Miami, Fla., and R:base from Microrim, of Bellevue, Wash., provide access **locking** down to the field **level** instead of the record **level** .
Deadlock facility. This requirement is a result of certain record-lockout techniques that, in some circumstances, can indefinitely freeze the **database** -management system. A species of software crashing known as "deadly embrace' can occur when two...

17/3,K/37 (Item 37 from file: 275)
DIALOG(R)File 275:Gale Group Computer DB(TM)

01208707 SUPPLIER NUMBER: 05031354 (USE FORMAT 7 OR 9 FOR FULL TEXT)
The DOS-LAN juncture. (designing and maintaining DOS applications for LANs)
Haugdahl, J. Scott
PC Tech Journal, v5, n7, p78(9)
July, 1987
ISSN: 0738-0194 LANGUAGE: ENGLISH RECORD TYPE: FULLTEXT; ABSTRACT
WORD COUNT: 6241 LINE COUNT: 00477

... application.
Even developers who do not have access to a LAN can experiment with the **file** -sharing and region-locking features of DOS on a stand-alone PC. SHARE.EXE, a...

...function as network servers. It performs equally well as a test bed for exploring LAN **file** -access questions.

Deadlocks . In any **kind** of resource **locking** scheme, an application must be prepared for **deadlock** . This circular wait condition develops as follows: application A has a lock on record 100 in **database** X and application B has a lock on record 150, also in **database** X. To complete its transaction, application A needs to update information in record 150 that...

17/3,K/38 (Item 38 from file: 275)
DIALOG(R)File 275:Gale Group Computer DB(TM)

01197713 SUPPLIER NUMBER: 06159780
The performance of alternative strategies for dealing with deadlocks in database management systems.
Agrawal, Rakesh; Carey, Michael J.; McVoy, Lawrence W.
IEEE Transactions on Software Engineering, v13, n12, p1348(16)
Dec, 1987
ISSN: 0098-5589 LANGUAGE: ENGLISH RECORD TYPE: ABSTRACT

...ABSTRACT: Locking seems to be the concurrency control strategy of choice for a wide variety of **database** workloads and system configurations, but locking opens the possibility of deadlocks. The strategies studied are based on **deadlock** detection, **deadlock** prevention, or timeouts. The choice of best **deadlock** resolution strategy depends on the **level** of data contention, the resource utilization levels, and the types of transactions.

17/3,K/39 (Item 39 from file: 275)
DIALOG(R)File 275:Gale Group Computer DB(TM)

01049581 SUPPLIER NUMBER: 00540663
Concurrency Control in Admin.
Parimala, N.; Prakash, N.; Bolloju, N.
Computer Journal, v27, n1, p62-66
Feb., 1984
ISSN: 0010-4620 LANGUAGE: ENGLISH RECORD TYPE: ABSTRACT

ABSTRACT: Concurrency control in **database** systems has to ensure that each user sees a consistent view of that **database** . Locking units are udes by most systems to ensure consistency. **Deadlocks** and **level** of concurrency in avoiding **deadlocks** are issues in concurrency control. A method using two resource lists in scheduling multiple users...

17/3,K/40 (Item 1 from file: 636)
DIALOG(R)File 636:Gale Group Newsletter DB(TM)

02548739 Supplier Number: 45140477 (USE FORMAT 7 FOR FULLTEXT)
IBM TOOLS REDUCE COMPLEXITY OF MULTI-VENDOR DATABASE ENVIRONMENTS
M2 Presswire, pN/A
Nov 15, 1994
Language: English Record Type: Fulltext
Document Type: Newswire; Trade
Word Count: 1246

... use graphical manner, instantly allowing customers to use this information to tune SQL statements and **databases** for better performance and efficiency.

DB2 Visual Explain also provides users with a "what-if...

...capability for SQL statements, allowing them to model the impact of

various changes in the **database** environment.

DB2 Performance Monitor provides performance data collection, viewing, reporting, analysis and alerting for the DB2/2 and DB2/6000 **databases** . This product supports more than 200 performance attributes, including buffer pool, lock and **deadlock** , sorting, communication, agent and **logging** information.

A variety of reporting methods are supported, with filtering and sorting capabilities to help...

...with Version 2.0 of DB2/2 and DB2/6000, enables a single view of **database** and system management. with the SNMP agent, systems management products such as IBM NetView* can be used to query and monitor **database** status configuration parameters and performance attributes, such as disk and logical reads and writes.

Data...

17/3,K/41 (Item 2 from file: 636)

02509520 Supplier Number: 45045335 (USE FORMAT 7 FOR FULLTEXT)

INTEGRATED DATABASE SYSTEMS MANAGEMENT TOOLS FOR AIX, OS/2 ANNOUNCED

Report on IBM, v11, n39, pN/A
Oct 5, 1994
Language: English Record Type: Fulltext
Document Type: Newsletter; Trade
Word Count: 779

... use graphical manner, instantly allowing customers to use this information to tune SQL statements and **databases** for better performance and efficiency.

DB2 Visual Explain users also have a "what-if" modeling capability for SQL statements which lets them model the impact of various changes in the **database** environment.

DB2 Performance Monitor collects, views, reports, analyzes, and sends alerts on performance data for DB2/2 and DB2/6000 **databases** . The product supports more than 200 performance attributes, including buffer pool, lock, and **deadlock** , sorting, communication, agent, and **logging** **information** .

DB2 Performance **Monitor** supports a variety of reporting methods along with filtering and sorting capabilities to help analyze...
...exception monitoring, letting users define threshold values.

The SNMP Agent enables a single view of **database** and systems management. With the SNMP Agent, systems management products such as IBM NetView can be used to query and monitor **database** status configuration parameters and performance attributes, such as disk and logical reads and writes.

IBM...

17/3,K/42 (Item 3 from file: 636)

02508836 Supplier Number: 45043563 (USE FORMAT 7 FOR FULLTEXT)

IBM INTRODUCES INTEGRATED DATABASE SYSTEMS MANAGEMENT TOOLS FOR AIX, OS/2

Network Management Systems & Strategies, v6, n19, pN/A
Oct 4, 1994
Language: English Record Type: Fulltext
Document Type: Newsletter; Trade
Word Count: 777

... use graphical manner, instantly allowing customers to use this information to tune SQL statements and **databases** for better performance and efficiency.

DB2 Visual Explain users also have a "what-if" modeling capability for SQL statements which lets them model the impact of various changes in the **database** environment.

DB2 Performance Monitor collects, views, reports, analyzes, and sends alerts on performance data for DB2/2 and DB2/6000 **databases** . The product supports more than 200 performance attributes, including buffer pool, lock, and **deadlock** , sorting, communication, agent, and **logging information** .

DB2 Performance **Monitor** supports a variety of reporting methods along with filtering and sorting capabilities to help analyze...
...exception monitoring, letting users define threshold values.

The SNMP Agent enables a single view of **database** and systems management. With the SNMP Agent, systems management products such as IBM NetView can be used to query and monitor **database** status configuration parameters and performance attributes, such as disk and logical reads and writes.

IBM...

17/3,K/43 (Item 4 from file: 636)
DIALOG(R)File 636:Gale Group Newsletter DB(TM)

02173610 Supplier Number: 44088687 (USE FORMAT 7 FOR FULLTEXT)
SCHOOF FILES FOR ARBITRATION AGAINST ARTISOFT
Computergram International, n2251, pN/A
Sept 10, 1993
Language: English Record Type: Fulltext
Document Type: Newswire; Trade
Word Count: 86

(USE FORMAT 7 FOR FULLTEXT)
TEXT:
John Schoof, ousted earlier this year as chief executive of Inc, Tucson, Arizona has **filed** a demand for arbitration and claims $1.5m in salary and bonus and vesting of...

...the Artisoft board, two to replace vacancies, and a third to prevent the chance of **deadlock** because the board currently **numbers** six people; Schoof has **retained** his 21% stake in the firm, giving him the right to make the nominations.

17/3,K/44 (Item 5 from file: 636)
DIALOG(R)File 636:Gale Group Newsletter DB(TM)

01500993 Supplier Number: 42113156 (USE FORMAT 7 FOR FULLTEXT)
MicroSTEP, VERSION 1.5
CASE Strategies, v3, n6, pN/A
June, 1991
Language: English Record Type: Fulltext
Document Type: Magazine/Journal; Trade
Word Count: 4186

... user-created backup batch file. Menu item picks initiate execution of .EXE, .COM, or .BAT **files** . MPL menus require about 2.5K of user memory, or considerably more if a .BAT **file** is chosen and the operating system is a version below DOS 4.

There are also utilities for importing existing Btrieve data **files** for use in MicroSTEP applications.

For multiuser applications, MicroSTEP provides for **locking** at the record **level** . **Deadlocks** (which occur when two users are each waiting for the other to release a record...

17/3,K/45 (Item 1 from file: 16)
DIALOG(R)File 16:Gale Group PROMT(R)

05991814 Supplier Number: 53361247 (USE FORMAT 7 FOR FULLTEXT)
Enterprise Solutions -- The Bridge Over Troubled Waters.(Transamerica Flood

Hazard Certification's [redacted]ve to Microssoft SQL Server 7 D[redacted])(Company Operations)
Hersch, Warren S.
Computer Reseller News, p2(1)
Dec 7, 1998
Language: English Record Type: Fulltext
Document Type: Magazine/Journal; Trade
Word Count: 733

... TA Flood and PCSI, a key issue was the performance of SQL 7's row-**level - locking** feature. Using parallel processing, the feature prevents **deadlocking** , a **database** freeze that occurs when two call-center representatives simultaneously update the same record. SQL 6.5 offered a more narrow page-level-locking capability.
Also at issue was the **database** 's compatibility with Delphi 4, which PCSI used to create the **database** application suite.
"When we tested in October, Inprise had not yet released a SQL 7...

17/3,K/46 (Item 2 from file: 16)
DIALOG(R)File 16:Gale Group PROMT(R)

03006336 Supplier Number: 44080590 (USE FORMAT 7 FOR FULLTEXT)
Verification dons formal attire: Mathematical tools promise to speed design
Electronic Engineering Times, p1
Sept 6, 1993
Language: English Record Type: Fulltext
Document Type: Magazine/Journal; Trade
Word Count: 1819

... make it possible to take a high-level specification and test for properties such as **deadlock** in a **cache** system.
That **type** of verification, which may play a strong role in the emerging electronic system design automation...

17/3,K/47 (Item 3 from file: 16)
DIALOG(R)File 16:Gale Group PROMT(R)

02667887 Supplier Number: 43556702 (USE FORMAT 7 FOR FULLTEXT)
Application Software
VARbusiness, p60
Jan, 1993
Language: English Record Type: Fulltext
Document Type: Magazine/Journal; Trade
Word Count: 214

... Designed for use in a multiuser environment and multiple-user counts, the product provides automatic **file** and **record** locking, which ensures **data** integrity. It also prevents **deadlock** when multiple users are accessing the same data. Paradox refreshes each user's screen as...

17/3,K/48 (Item 1 from file: 148)
DIALOG(R)File 148:Gale Group Trade & Industry DB

07995464 SUPPLIER NUMBER: 17278550 (USE FORMAT 7 OR 9 FOR FULL TEXT)
Informix Software Inc. Endorses BMC Software's PATROL to Provide Database Management for INFORMIX-OnLine Dynamic Server Architecture Databases.
Business Wire, p7180067
July 18, 1995
LANGUAGE: English RECORD TYPE: Fulltext
WORD COUNT: 781 LINE COUNT: 00084

... from leading systems, network, hardware and tool vendors.

Joint Technology Programs

Today, PATROL supports Informix **databases** through the Knowledge Module for Informix, a library of expertise that contains rules for how PATROL will manage Informix **databases** . It provides customized management features specific to OnLine Dynamic Server and other **databases** based on Informix Dynamic Scalable Architecture (DSA). These customized features support **database** administration tasks as well as extensive monitoring. The Knowledge Module for Informix monitors critical Informix parameters such as table scans, **deadlocks** , buffer waits and overflows, **cache** hit rates and more than 70 other relevant parameters. When parameters that exceed defined exception...
...soon to be released SNMP-based agent, allowing for an even more robust set of **database** metrics.

Pricing and Availability

PATROL is available today from BMC Software and its agents and...

17/3,K/49 (Item 2 from file: 148)
DIALOG(R)File 148:Gale Group Trade & Industry DB

07570964 SUPPLIER NUMBER: 16250825 (USE FORMAT 7 OR 9 FOR FULL TEXT)
Databases conquer new LANs. (Microsoft Corp's SQL Server for Windows NT 4.21A, Oracle Corp's Oracle 7.0.16 database application development software packages) (includes related articles on evaluation summary, ratings guide, database security, software documentation, system requirements, sequential searches, cost to build databases, page-level locking, testing platform) (Software Review) (Evaluation)
Watt, Peggy; Dowgiallo, Edward; Johnson, Amy H.
InfoWorld, v16, n46, p128(9)
Nov 14, 1994
DOCUMENT TYPE: Evaluation ISSN: 0199-6649 LANGUAGE: ENGLISH
RECORD TYPE: FULLTEXT; ABSTRACT
WORD COUNT: 10072 LINE COUNT: 00787

... capable of finishing its transaction while the other waited.

The advent of page- and row- **level locking** in SQL **database** systems lets **deadlocks** now appear within tables (see diagram). **Deadlocks** occur more frequently if the **database** locks pages instead of rows.

What makes deadlocks nasty is that the proper way to...

17/3,K/50 (Item 1 from file: 15)
DIALOG(R)File 15:ABI/Inform(R)

02162834 73016489
High speed transaction recovery
Mullins, Craig S
Computer Technology Review PP: 44-52 First Quarter 2001
ISSN: 0278-9647 JRNL CODE: CTN
WORD COUNT: 4464

...TEXT: analysis process. Or it is possible that there has been even more activity on the **database** after the analysis and new anomalies have appeared. A minimum set of capabilities would include:

* Retry on **deadlock** situations.

* The ability to **log** and defer SQL statements with problems (e.g. inserts where the key is already in...

17/3,K/51 (Item 2 from file: 15)
DIALOG(R)File 15:ABI/Inform(R)

01741813 03-92803

The bridge over troubled ters
Hersch, Warren S
Computer Reseller News n820 PP: 2, 8 Dec 7, 1998
ISSN: 0893-8377 JRNL CODE: CRN
WORD COUNT: 698

...TEXT: TA Flood and PCSI, a key issue was the performance of SQL 7's row- **level - locking** feature. Using parallel processing, the feature prevents **deadlocking** , a **database** freeze that occurs when two callcenter representatives simultaneously update the same record. SQL 6.5 offered a more narrow pagelevel-locking capability.

Also at issue was the **database** 's compatibility with Delphi 4, which PCSI used to create the **database** application suite.

"When we tested in October, Inprise had not yet released a SQL 7...

17/3,K/52 (Item 3 from file: 15)
DIALOG(R)File 15:ABI/Inform(R)

00872206 95-21598
Distributed not yet delivered
McGoveran, David
Computerworld v28n23 PP: 112, 114 Jun 6, 1994
ISSN: 0010-4841 JRNL CODE: COW
WORD COUNT: 2281

...TEXT: area. This simply evades the issue of how difficult it is to manage a distributed **database** environment when systems administration and security facilities are incapable of treating a distributed **database** as a single logical entity.

Failed distributed transactions are particularly problematic in today's products...

... recovering failed transactions are still primitive, often requiring detailed manual analysis of transaction and error **log** entries.

Deadlocks in a distributed environment can cause severe system degradation. Provisions must also be made for either global **deadlock** avoidance or global **deadlock** detection and recovery.
Vendors usually implement a time-out mechanism, incorrectly referring to it as a means of **deadlock** avoidance; it is not because it only ends a **deadlock** if one happens to exist, and a time-out cannot differentiate between long-running queries and a true **deadlock** condition.

Few, if any, **DBMS** products handle distributed **deadlock** avoidance, detection or recovery.

7. AVAILABILITY AND AUTONOMY

Local autonomy does not mean the local...

... remote site. Instead this delegation is an efficiency measure that should not prevent the local **database** from continuing operation in case it becomes disconnected from remote sites.

It should always be...

17/3,K/53 (Item 4 from file: 15)
DIALOG(R)File 15:ABI/Inform(R)

00701207 93-50428
Relational DBMS servers: Cutting through the thicket

Li, Paul; Wendel, Eric; H[illegible]d, Jeff
Network World v10n18 PP: 34-39 May 3, 1993
ISSN: 0887-7661 JRNL CODE: NWW
WORD COUNT: 3481

...TEXT: concurrency.

Databases support certain degrees of locking granularity; the higher the granularity, the faster the **database** performs. While most of the vendors were able to provide a high level of granularity...

... able to provide row-level locking. The three firms only go as far as table- **level locking** .

Two related core features are also worth consideration: autolock escalation and **deadlock** resolution.

Autolock escalation is the ability to aggregate multiple locked pages of data into a...
...autolock escalation may have an adverse effect on concurrency by locking up more of the **database** than is necessary, it often results in decreased delay to end

17/3,K/54 (Item 5 from file: 15)
DIALOG(R)File 15:ABI/Inform(R)

00653415 93-02636
Guide to Strategic Purchase Decisions: Distributed DBMS - The Elusive Illusion
Nitzsche, Kyle
Network World v9n48 PP: 43-44, 60, 63 Nov 30, 1992
ISSN: 0887-7661 JRNL CODE: NWW
WORD COUNT: 3858

...TEXT: integrity of data so that tables on distant nodes that are dependent on a master **database** record are automatically updated when the master record is updated.

* Local autonomy. With this element, **database** administrators retain control over the data stored at their site.

* Continual operation. This is a...

...to another without affecting application processing.

* Distributed transaction management. This feature enables recovery of a **database** spanning multiple nodes in compliance with any existing integrity constraints.

* Global concurrency and **deadlock** recovery. This manages locks on **data records** across a distributed **database** . It also detects and resolves any conditions where two nodes may be **deadlocked** in their efforts to access each other's data.

* Hardware, software, network and third-party **DBMS** independence via built-in support or gateways. These elements fit the distributed **DBMS** into existing environments and data systems.

Nitzsche is an associate features editor for Network World.

17/3,K/55 (Item 6 from file: 15)
DIALOG(R)File 15:ABI/Inform(R)

00461830 89-33617

Concurrency Control in Distributed Databases Through Time Intervals and Short-Term Locks
Halici, Ugur; Dogac, Asuman
IEEE Transactions on Software Engineering v15n8 PP: 994-1003 Aug 1989
ISSN: 0098-5589 JRNL CODE: ISO

...ABSTRACT: the ordering by serialization numbers (OSN) method, is proposed for providing concurrency control in distributed **database** management systems. The OSN method, which increases the level of concurrent execution of transactions, works...

... mode and uses time interval techniques in conjunction with short-term locks to provide serializability. **Deadlocks** are prevented by the technique. The scheduler is distributed, and the standard transaction execution policy is assumed. The OSN method provides more concurrency than basic timestamp ordering and 2-phase **locking** methods and successfully handles some **logs** that cannot be handled by any previous methods. A complexity analysis of the algorithm indicates...

17/3,K/56 (Item 1 from file: 647)
DIALOG(R)File 647:CMP Computer Fulltext

01180082 CMP ACCESSION NUMBER: CRN19981207S0020
Enterprise Solutions - The Bridge Over Troubled Waters
Warren S. Hersch
COMPUTER RESELLER NEWS, 1998, n 820, PG2
PUBLICATION DATE: 981207
JOURNAL CODE: CRN LANGUAGE: English
RECORD TYPE: Fulltext
SECTION HEADING: News
WORD COUNT: 727

... TA Flood and PCSI, a key issue was the performance of SQL 7's row- **level** - **locking** feature. Using parallel processing, the feature prevents **deadlocking** , a **database** freeze that occurs when two call- center representatives simultaneously update the same record. SQL 6.5 offered a more narrow page-level-locking capability.

Also at issue was the **database** 's compatibility with Delphi 4, which PCSI used to create the **database** application suite.

"When we tested in October, Inprise had not yet released a SQL 7...

17/3,K/57 (Item 2 from file: 647)
DIALOG(R)File 647:CMP Computer Fulltext

00640200 CMP ACCESSION NUMBER: IWK19891127S0739
Juggling DBMSs - IS executives who rushed to IBM's DB2 for fear of being left behind now find themselves facing a dilemma
David Kull
INFORMATIONWEEK, 1989, n 248, 41
PUBLICATION DATE: 891127
JOURNAL CODE: IWK LANGUAGE: English
RECORD TYPE: Fulltext
SECTION HEADING: PRODUCT FOCUS
WORD COUNT: 2072

... those systems. Neither Hewett nor Northern States, for example, plans large- scale conversions of existing **databases** to DB2, and they continue to develop new applications in Datacom/DB and Adabas, respectively...

...s record-level concurrency-control locking as one significant advantage over DB2, which uses page- **level** **locking** . At the lower **level** of

detail, this allows for greater transaction concurrency and fewer troublesome **deadlocks** .

Paske says that in many cases an application might be handled by either system. He...

17/3,K/58 (Item 3 from file: 647)
DIALOG(R)File 647:CMP Computer Fulltext

00551597 CMP ACCESSION NUMBER: VAR19930101S9980
Application Software
VARBUSINESS, 1993, n 901 , 60
PUBLICATION DATE: 930101
JOURNAL CODE: VAR LANGUAGE: English
RECORD TYPE: Fulltext
SECTION HEADING: TOP PRODUCTS OF THE YEAR
WORD COUNT: 211

... Designed for use in a multiuser environment and multiple- user counts, the product provides automatic **file** and **record** locking, which ensures **data** integrity. It also prevents **deadlock** when multiple users are accessing the same data. Paradox refreshes each user's screen as...

17/3,K/59 (Item 4 from file: 647)
DIALOG(R)File 647:CMP Computer Fulltext

00547062 CMP ACCESSION NUMBER: OST19930705S5420
Data Warehouses Make Sense When Multiple Users Need Data Access (Data Management)
Julie Anderson
OPEN SYSTEMS TODAY, 1993, n 127, 26
PUBLICATION DATE: 930705
JOURNAL CODE: OST LANGUAGE: English
RECORD TYPE: Fulltext
SECTION HEADING: Development Tools
WORD COUNT: 575

... loads because the data are static.

The data server requirements also are different from operational **databases** . The only operation that the **data warehouse** server has to support is a read. Updates, adds and deletes are not necessary-once the data are loaded correctly. This eliminates the need for exclusive **locks** , **deadlock** protection, recovery **logs** and the other overhead of OLTP **DBMS** systems.

As new data are loaded, older data can be archived, summarized or deleted. Thus, details of daily sales for one month may be stored in the **data warehouse** . At the end of the month, only monthly summaries need to be stored in the...

...an optical or tape archive.

Since the data are not straight copies of the operational **databases** , the **data warehouse** needs to provide a description of the data to the decision support user. This is...

```
File   8:Ei Compendex(R) 1970-2004/Jul W4
         (c) 2004 Elsevier Eng.  Info. Inc.
File  35:Dissertation Abs Online 1861-2004/May
         (c) 2004 ProQuest Info&Learning
File 202:Info. Sci. & Tech. Abs. 1966-2004/Jul 12
         (c) 2004 EBSCO Publishing
File  65:Inside Conferences 1993-2004/Aug W1
         (c) 2004 BLDSC all rts. reserv.
File   2:INSPEC 1969-2004/Jul W4
         (c) 2004 Institution of Electrical Engineers
File  94:JICST-EPlus 1985-2004/Jul W2
         (c)2004 Japan Science and Tech Corp(JST)
File 483:Newspaper Abs Daily 1986-2004/Jul 31
         (c) 2004 ProQuest Info&Learning
File   6:NTIS 1964-2004/Aug W1
         (c) 2004 NTIS, Intl Cpyrght All Rights Res
File 144:Pascal 1973-2004/Jul W4
         (c) 2004 INIST/CNRS
File 434:SciSearch(R) Cited Ref Sci 1974-1989/Dec
         (c) 1998 Inst for Sci Info
File  34:SciSearch(R) Cited Ref Sci 1990-2004/Jul W4
         (c) 2004 Inst for Sci Info
File  99:Wilson Appl. Sci & Tech Abs 1983-2004/Jun
         (c) 2004 The HW Wilson Co.
File 266:FEDRIP 2004/Jun
         Comp & dist by NTIS, Intl Copyright All Rights Res
File  95:TEME-Technology & Management 1989-2004/Jun W1
         (c) 2004 FIZ TECHNIK
File 438:Library Lit. & Info. Science 1984-2004/Jun
         (c) 2004 The HW Wilson Co

Set     Items   Description
S1      17396   DEADLOCK??? OR DEAD()LOCK???
S2       6954   (S1 OR LOCK???)(5N)(LEVEL? ? OR TIER? ? OR GRADE? ? OR STR-
             ENGTH OR DEGREE? ? OR KIND? ? OR TYPE? ? OR CATEGOR? OR CLASS-
             ?? OR CLASSIFICATION OR FAMILY OR FAMILIES)
S3        637   (S1 OR LOCK???)(5N)(HISTORY OR HISTORIES OR LOG? ? OR LOGG-
             ??? OR CACHE? ?)
S4        790   S1(5N)(DATA OR INFORMATION OR NUMBER? ? OR AMOUNT? ? OR ST-
             ATISTIC?? OR PROFILE? ?)
S5         24   S4(5N)(RETAIN??? OR STOR??? OR RECORD??? OR SAV??? OR MAIN-
             TAIN??? OR KEEP??? OR KEPT OR TRACK??? OR MONITOR???)
S6    1058088   DATABASE? ? OR DATA()(BASE? ? OR WAREHOUSE? ?) OR DBMS OR -
             RDBMS OR REPOSITOR??? OR (DATA OR INFORMATION)()MANAGEMENT OR
             FILE? ?
S7          1   S1 AND S2 AND (S3 OR S5) AND S6
S8          3   S1 AND S2 AND (S3 OR S5)
S9        125   S1 AND S2 AND S6
S10      4276   (S1 OR LOCK???)(5N)(LEVEL? ? OR TIER? ? OR KIND? ? OR TYPE?
              ?)
S11        76   S1 AND S10 AND S6
S12        55   S1(20N)S10 AND S6
S13        29   S1 AND (S3 OR S5) AND S6
S14        86   S7:S8 OR S12:S13
S15        68   RD (unique items)
S16        65   S15 NOT PY=2003:2004
S17       165   S1 AND (GRAIN??? OR GRANULAR?)
S18        17   S17 AND (S2 OR S3 OR S5)
S19        13   RD (unique items)
S20         8   S19 NOT S16
```

16/5/3 (Item 3 from file: 8)
DIALOG(R)File 8:Ei Compendex(R)

03603281 E.I. Monthly No: EIM9305-024772
Title: Transaction management for global serializability and local autonomy in multidatabase systems.
Author: Hwang, Buhyun; Moon, Songchun
Corporate Source: Chonnam Natl Univ, Kwangju, South korea
Conference Title: 18th EUROMICRO Symposium on Microprocessing and Microprogramming - EUROMICRO 92
Conference Location: Paris, Fr Conference Date: 19920914
E.I. Conference No.: 17255
Source: Microprocessing and Microprogramming v 35 n 1-5 Sep 1992. p 437-444
Publication Year: 1992
CODEN: MMICDT ISSN: 0165-6074
Language: English
Document Type: JA; (Journal Article) Treatment: T; (Theoretical); M; (Management Aspects)
Journal Announcement: 9305
Abstract: Global serializability and local autonomy in the MDBS (multidatabase system) are important issues since MDBS has been studied for the past five years. This paper proposes the transaction processing model for an MDBS and its three-level scheduling algorithm for global schedulers, for integrated local and global transaction schedulers (ILGSs), and for local schedulers to achieve global serializability, local autonomy, and higher concurrency. The most difficult problem to ensure global serializability for an MDBS is to schedule global transactions so that their execution order and their serialization order at each site are identical. In our algorithm, this problem is resolved by two different ILGS modules: the first ILGS module is used in the case that local concurrency control (LCC) algorithm produces a history in which the serialization point of a transaction is the same as its serializable order, and the second ILGS module is used for each LCC algorithm that generates a serializable history. To achieve a high degree of concurrency, transactions are scheduled at the ILGS module, regardless of they are local or global, in a way that unnecessary abort or delay is not caused. By simply adding the ILGS level on top of each local **DBMS** , local DBMSs need not be changed at all. Therefore, local autonomy is guaranteed. Our algorithm also provides freedom from a **deadlock** at the global **level** and may be used even if data items are replicated at many sites. (Author abstract) 18 Refs.
Descriptors: **DATABASE** SYSTEMS; SCHEDULING; ALGORITHMS; SUPERVISORY AND EXECUTIVE PROGRAMS; COMPUTER SYSTEM RECOVERY; MANAGEMENT
Identifiers: SERIALIZATION POINT; GLOBAL SERIALIZABILITY; LOCAL AUTONOMY; LOCAL CONCURRENCY CONTROL
Classification Codes:
723 (Computer Software); 722 (Computer Hardware)
72 (COMPUTERS & DATA PROCESSING)

16/5/4 (Item 4 from file: 8)
DIALOG(R)File 8:Ei Compendex(R)

03022151 E.I. Monthly No: EIM9102-004808
Title: Reliable transaction management in a multidatabase system.
Author: Breitbart, Yuri; Silberschatz, Avi; Thompson, Glenn R.
Corporate Source: Univ of Kentucky, Lexington, KY, USA
Conference Title: Proceedings of the 1990 ACM SIGMOD International Conference on Management of Data
Conference Location: Atlantic City, NJ, USA Conference Date: 19900523
Sponsor: ACM SIGMOD, New York, NY, USA
E.I. Conference No.: 13639
Source: SIGMOD Record (ACM Special Interest Group on Management of Data) v 19 n 2. Publ by ACM, Fort Collins Computer Center, Fort Collins, CO, USA. p 215-224
Publication Year: 1990

CODEN: SRECD8 ISSN: 0163-5808
Language: English
Document Type: PA; (Conference Paper) Treatment: T; (Theoretical); X; (Experimental)
Journal Announcement: 9102
Abstract: A model of a multidatabase system is defined in which each local **DBMS** uses the two-phase locking protocol. Locks are released by a global transaction only after the transaction commits or aborts at each local site. Failures may occur during the processing of transactions. We design a fault tolerant transaction management algorithm and recovery procedures that retain global **database** consistency. We also show that our algorithms ensure freedom from global **deadlocks** of any **kind** . (Author abstract) 14 Refs.
Descriptors: **DATABASE** SYSTEMS--*Management; COMPUTER PROGRAMMING--Algorithms; COMPUTER NETWORKS--Protocols
Identifiers: MULTIDATABASE SYSTEMS
Classification Codes:
723 (Computer Software)
72 (COMPUTERS & DATA PROCESSING)

16/5/5 (Item 5 from file: 8)
DIALOG(R)File 8:Ei Compendex(R)

02833075 E.I. Monthly No: EI8912128276
Title: Concurrency control in distributed databases **through time intervals and short-term locks.**
Author: Halici, Ugur; Dogac, Asuman
Corporate Source: Middle East Technic Univ, Dep of Electr & Electron Eng, Ankara, Turk
Source: IEEE Transactions on Software Engineering v 15 n 8 Aug 1989 p 994-1003
Publication Year: 1989
CODEN: IESEDJ ISSN: 0098-5589
Language: English
Document Type: JA; (Journal Article) Treatment: T; (Theoretical); A; (Applications)
Journal Announcement: 8912
Abstract: A method for concurrency control in distributed **database** management systems that increases the level of concurrent execution of transactions, called ordering by serialization numbers (OSN), is proposed. The OSN method works in the certifier mode and uses time-interval techniques in conjunction with short-term locks to provide serializability and prevent **deadlocks** . The scheduler is distributed, and the standard transaction execution policy is assumed, that is, the read and write operations are issued continuously during transaction execution. However, the write operations are copied into the **database** only when the transaction commits. The amount of concurrency provided by the OSN method is demonstrated by log classification. It is shown that the OSN method provides more concurrency than basic timestamp ordering and two-phase **locking** methods and handles successfully some **logs** which cannot be handled by any of the past methods. The complexity analysis of the algorithm indicates that the method works in a reasonable amount of time. 22 Refs.
Descriptors: **DATABASE** SYSTEMS--*Distributed; COMPUTER SYSTEMS PROGRAMMING--Multiprocessing Programs
Identifiers: CONCURRENCY CONTROL; TRANSACTION PROCESSING SYSTEMS
Classification Codes:
723 (Computer Software)
72 (COMPUTERS & DATA PROCESSING)

16/5/7 (Item 7 from file: 8)
DIALOG(R)File 8:Ei Compendex(R)

02549862 E.I. Monthly No: EI8804032694

Title: **PERFORMANCE OF ALTERNATIVE STRATEGIES FOR DEALING WITH DEADLOCKS IN** DATABASE **MANAGEMENT SYSTEMS.**
Author: Agrawal, Rakesh; Carey, Michael J.; McVoy, Lawrence W.
Corporate Source: AT&T Bell Lab, Murray Hill, NJ, USA
Source: IEEE Transactions on Software Engineering v SE-13 n 12 p 1348-1363
CODEN: IESEDJ ISSN: 0098-5589
Language: ENGLISH
Treatment: T; (Theoretical); X; (Experimental)
Journal Announcement: 8804
Abstract: The authors use a simulation model of a **database** environment to study the relative performance of several strategies based on deadlock detection, several strategies based on **deadlock** prevention, and a strategy based on timeouts. It is shown that the choice of the best **deadlock** resolution strategy depends on the **level** of data contention, the resource utilization levels, and the types of transactions. Guidelines are provided for selecting a **deadlock** resolution strategy for different operating regions. 41 refs.
Descriptors: **DATABASE** SYSTEMS; PROBABILITY--Queueing Theory
Identifiers: **DATABASE** MANAGEMENT SYSTEMS; DEADLOCK DETECTION; DEADLOCK RESOLUTION STRATEGY; CONCURRENCY CONTROL; TRANSACTION PROCESSING
Classification Codes:
723 (Computer Software); 922 (Statistical Methods)
72 (COMPUTERS & DATA PROCESSING); 92 (ENGINEERING MATHEMATICS)

16/5/8 (Item 8 from file: 8)
DIALOG(R)File 8:Ei Compendex(R)

02264691 E.I. Monthly No: EIM8708-051941
Title: FINER GRAINED CONCURRENCY FOR THE DATABASE **CACHE.**
Author: Moss, J. Eliot B.; Leban, Bruce; Chrysanthis, Panos K.
Corporate Source: Univ of Massachusetts, Amherst, MA, USA
Conference Title: Proceedings - Third International Conference on Data Engineering.
Conference Location: Los Angeles, CA, USA Conference Date: 19870203
Sponsor: IEEE Computer Soc, Los Alamitos, CA, USA
E.I. Conference No.: 09779
Source: Publ by IEEE, New York, NY, USA. Available from IEEE Service Cent (Cat n 87CH2407-5), Piscataway, NJ, USA p 96-103
Publication Year: 1987
ISBN: 0-8186-0762-9
Language: English
Document Type: PA; (Conference Paper)
Journal Announcement: 8708
Abstract: The **database** cache transaction recovery technique offers significant performance advantages for reliable **database** systems. However, the smallest granularity of locks it provides is the page. The authors present two schemes supporting smaller granularity. The first scheme allows maximal concurrency consistent with physical two-phase locking, with the same per-transaction I/O cost as the original **deadlock cache** scheme. The second scheme offers the same concurrency as the first, but features reduced I/O on commit, at the cost of some increase in recovery time. 4 refs.
Descriptors: **DATABASE** SYSTEMS--*Performance; COMPUTER SYSTEMS PROGRAMMING--Input Output Programs
Identifiers: **DATABASE CACHE** ; TWO-PHASE **LOCKING** ; **DEADLOCK CACHE** SCHEME
Classification Codes:
723 (Computer Software)
72 (COMPUTERS & DATA PROCESSING)

16/5/10 (Item 10 from file: 8)
DIALOG(R)File 8:Ei Compendex(R)

02192907 E.I. Monthly No: EI8704034271
Title: DEADLOCK **DETECTION ALGORITHM WITH** LEVEL **NUMBER.**
Author: Eguchi, Yoshikazu; Yoshinaga, Tsunehiro
Corporate Source: Tokuyama Technical College, Tokuyama, Jpn
Source: Syst Comput Jpn v 17 n 11 Nov 1986 p 42-50
Publication Year: 1986
CODEN: SCJAEP ISSN: 0882-1666
Language: ENGLISH
Document Type: JA; (Journal Article) Treatment: T; (Theoretical)
Journal Announcement: 8704
Abstract: This paper proposes an efficient algorithm for detecting the deadlock in the concentrated **database** system, and demonstrates its correctness. The algorithm works as follows. When a process makes an access to a resource which is locked by another process, the process must wait for the resource. The algorithm detects the deadlock by tracing the edges of the resource graph, starting from the waiting process. By attaching the level number to each node, the number of edge tracings can be reduced, which helps to reduce the overhead for deadlock detection. To evaluate the efficiency of the proposed algorithm, an experiment was performed algorithm, an experiment was performed by computer simulation. The cases of the algorithm with and without the proposed level number are compared, indicating the effectiveness of the proposed method. (Edited author abstract) 5 refs.
Descriptors: **DATABASE** SYSTEMS; COMPUTER PROGRAMMING--Algorithms; COMPUTER SIMULATION; MATHEMATICAL TECHNIQUES--Graph Theory
Identifiers: **DEADLOCK** DETECTION ALGORITHM; **LEVEL** NUMBER; RESOURCE GRAPH
Classification Codes:
723 (Computer Software); 921 (Applied Mathematics)
72 (COMPUTERS & DATA PROCESSING); 92 (ENGINEERING MATHEMATICS)

16/5/11 (Item 11 from file: 8)
DIALOG(R)File 8:Ei Compendex(R)

02117157 E.I. Monthly No: EIM8609-059900
Title: LOCAL DISTRIBUTED DEADLOCK DETECTION WITH FINITE BUFFERS.
Author: Cidon, Israel; Jaffe, Jeffrey M.; Sidi, Moshe
Corporate Source: Technion-Israel Inst of Technology, Haifa, Isr
Conference Title: Proceedings - IEEE INFOCOM '86, Fifth Annual Conference: Computers and Communications Integration Design, Analysis, Management.
Conference Location: Miami, FL, USA Conference Date: 19860408
Sponsor: IEEE Computer Soc, Los Alamitos, CA, USA; IEEE Communications Soc, New York, NY, USA
E.I. Conference No.: 08299
Source: Proceedings - IEEE INFOCOM 5th. Publ by IEEE, New York, NY, USA. Available from IEEE Service Cent (Cat n 86CH2284-8), Piscataway, NJ, USA p 478-487
Publication Year: 1986
CODEN: PINFEZ ISSN: 0743-166X ISBN: 0-8186-0694-0
Language: English
Document Type: PA; (Conference Paper)
Journal Announcement: 8609
Abstract: Most distributed algorithms to locate deadlocks (e. g. resource deadlocks in **database** systems) rely on learning the graph that represents which node is waiting for each other node. If the **deadlock** is a 'buffer **deadlock** ' at the network **level** , then there is insufficient storage in network nodes to learn this graph. The authors describe and validate a local distributed algorithm that finds buffer deadlocks using only a fixed number of buffers per node plus a fixed number of buffers per adjacent link. 17 refs.
Descriptors: **DATABASE** SYSTEMS--*Performance; COMPUTER PROGRAMMING--Algorithms
Identifiers: FINITE BUFFERS; DEADLOCKS
Classification Codes:
723 (Computer Software)

72 (COMPUTERS & DATA PROCESSING)

16/5/16 (Item 16 from file: 8)
DIALOG(R)File 8:Ei Compendex(R)

00865327 E.I. Monthly No: EI7912093860 E.I. Yearly No: EI79022108
Title: MULTI- LEVEL LOCKING **WITH** DEADLOCK **AVOIDANCE.**
Author: Lomet, David B.
Corporate Source: IBM, Thomas J. Watson Res Cent, Yorktown Heights, NY
Source: Proc Annu Conf ACM Washington, DC, Dec 4-6 1978. Publ by ACM, New York, NY, 1978 v 2 p 862-867
Publication Year: 1978
CODEN: PACMDC
Language: ENGLISH
Journal Announcement: 7912
Abstract: The existence of multiple levels of resources in **data base** systems presents a problem when resources must be shared among contending users. For example, some users may be interested in an entire **file** while other are only interested in one of a few records of the **file** . Intention locks have previously been introduced to permit multiple **levels** of **locking** in conjunction with **deadlock** detection methods. These intention locks can be viewed as housekeeping entries to record the pattern of usage of resources. When viewed in this way, no new locking modes need be exposed to a system user. Further, the basic idea behind intention locks can then be directly applied to systems which use deadlock avoidance method. 4 refs.
Descriptors: **DATA BASE** SYSTEMS
Classification Codes:
723 (Computer Software)
72 (COMPUTERS & DATA PROCESSING)

16/5/19 (Item 3 from file: 35)
DIALOG(R)File 35:Dissertation Abs Online

01097167 ORDER NO: AAD90-09679
DEADLOCK DETECTION AND RESOLUTION IN DATABASE **MANAGEMENT SYSTEMS: A COMPREHENSIVE APPROACH**
Author: PARK, YOUNG CHUL
Degree: PH.D.
Year: 1989
Corporate Source/Institution: NORTHWESTERN UNIVERSITY (0163)
ADVISER: PETER SCHEUERMANN
Source: VOLUME 50/11-B OF DISSERTATION ABSTRACTS INTERNATIONAL. PAGE 5176. 179 PAGES
Descriptors: COMPUTER SCIENCE
Descriptor Codes: 0984

In this dissertation, several algorithms for deadlock detection and resolution in **database** management systems are presented, where two-phase locking is assumed for ensuring serializability, the lock requests obey the granularity locking protocol and each granule may be locked in one of the following lock modes: IS, IX, S, SIX and X. For each object, lock requests are honored according to a first-come-first-served basis except for lock upgradations.

We present algorithms for deadlock detection and resolution in sequential transaction processing to achieve the goal of early deadlock detection with the appropriate victim selection. We also present a deadlock detection and resolution algorithm for parallel transaction processing which achieves the same objectives and an algorithm for distributed transaction processing which minimizes the amount of inter-site message communications.

We propose a new efficient algorithm for deadlock detection in sequential transaction processing, where the basic idea is the construction of a directed graph called a Holder/Walter-Transaction Waited-By Graph. We establish guidelines for the identification of a victim in a deadlock cycle

and show how deadlocks can be resolved with minimal victim cost. In addition, our algorithm allows us to resolve some **deadlocks** without aborting any transaction.

In the case of parallel transaction processing, a transaction can have multiple outstanding **lock** requests. We introduce two **types** of **deadlocks** : explicit **deadlocks** and implicit **deadlocks** . To detect **deadlocks** in this environment, we introduce a new type of directed graph called a transaction waited-by graph. We present deadlock detection mechanisms, identify deadlock detection time, and show how victims can be selected with minimal cost.

For deadlock detection and resolution in distributed **database** systems, the "traditional" transaction wait-for graph is augmented appropriately such that local deadlocks are detected locally by cycle detection and global deadlocks are detected globally by probe generation and propagation. A global deadlock is declared when a probe which was initiated by a global transaction returns to itself and that transaction becomes the victim for the resolution of the corresponding global deadlock. Antiprobes are generated appropriately to compensate the propagated probes.

16/5/20 (Item 4 from file: 35)
DIALOG(R)File 35:Dissertation Abs Online

831335 ORDER NO: AAD83-29811
CONCURRENCY AND TRANSACTION ROLLBACK IN DATABASE **SYSTEMS**
Author: BUCKLEY, GAEL NORMA
Degree: PH.D.
Year: 1983
Corporate Source/Institution: THE UNIVERSITY OF TEXAS AT AUSTIN (0227)
Source: VOLUME 44/09-B OF DISSERTATION ABSTRACTS INTERNATIONAL.
PAGE 2817. 66 PAGES
Descriptors: COMPUTER SCIENCE
Descriptor Codes: 0984

The use of multiprocessors and local area networks have greatly increased the number of simultaneous users of a **database** system. Such systems can improve their throughput and user turnaround time by allowing concurrent access to the **database** , but should still ensure that the results of each user computation are identical to the results when the computation is run alone on the **database** . This requirement is termed serializability. Many protocols have been devised to ensure serializability, and they cover a wide range of performance behaviors. One of the most important performances factors is the amount of transaction rollback in the system. Transaction rollback occurs when some number of user computations have to be undone and redone in order to ensure serializability. In this dissertation, I study several widely used **database** models to develop rollback-free and minimal rollback protocols for each model. Several of these protocols can be used to generate families of protocols suitable for restricted versions of the model.

The second part of this dissertation analyzes protocols which use locks to ensure serializability. All locking protocols use the locks to restrict access to an individual data item; however, some protocols extend the semantics to restrict access to an entire section of the **database** . This extension of lock semantics makes the various protocols difficult to compare, and reduces concurrency of the system. I define a new concept, the edge lock, to resolve this difficulty. It is used to enhance the concurrency of existing protocols, to simulate and integrate different **types** of **locking** protocols, and to illustrate the separate uses of locks for serializability and for **deadlock** -freedom.

16/5/21 (Item 5 from file: 35)
DIALOG(R)File 35:Dissertation Abs Online

776278 ORDER NO: AAD82-08220

STRATEGIES FOR ENHANCING CONCURRENCY AND MANAGING DEADLOCK IN DATA BASE LOCKING PROTOCOLS

Author: MOHAN, C.
Degree: PH.D.
Year: 1981
Corporate Source/Institution: THE UNIVERSITY OF TEXAS AT AUSTIN (0227)
Source: VOLUME 42/11-B OF DISSERTATION ABSTRACTS INTERNATIONAL.
PAGE 4487. 108 PAGES
Descriptors: COMPUTER SCIENCE
Descriptor Codes: 0984

With the ever growing popularity of **data base** management the systematic study of consistency preserving concurrency control techniques has become very important. Two important issues that need to be considered are: (i) the **level** of concurrency and (ii) **deadlocks** . This thesis is concerned with the problem of extending previous work on two-phase and non-two-phase locking protocols to achieve a higher degree of concurrency and at the same time deal effectively with the deadlock problem. Our work with the non-two-phase protocols deals with the most general of the existing natural protocols that are defined for use with **data bases** organized as directed acyclic graphs. An increased level of concurrency is attained by allowing lock conversions and/or by introducing new lock modes. When this is done either deadlock-freedom is assured a priori or simple restrictions are introduced to reduce the cost of deadlock detection and recovery.

In addition to extending existing proctocols and proposing new ones, we also extend the existing theory of locking protocols by including lock conversions and the new modes of locking in the directed hypergraph model of locking protocols. In so doing, we obtain very general results concerning serializability and deadlock-freedom properties of all protocols satisfying a natural closure property. We propose and use some new proof techniques.

16/5/22 (Item 6 from file: 35)
DIALOG(R)File 35:Dissertation Abs Online

754995 ORDER NO: AAD81-18667
LOCKING PROTOCOLS: GENERAL LOCK CLASSES AND DEADLOCK FREEDOM

Author: KORTH, HENRY FRANCIS
Degree: PH.D.
Year: 1981
Corporate Source/Institution: PRINCETON UNIVERSITY (0181)
Source: VOLUME 42/03-B OF DISSERTATION ABSTRACTS INTERNATIONAL.
PAGE 1085. 89 PAGES
Descriptors: COMPUTER SCIENCE
Descriptor Codes: 0984

This thesis is a study of the theory of lock modes in a **database** system. We first survey some of the issues of **database** concurrency in which locking plays a central role, including consistency, granularity, and deadlocks and waits. Rather than restricting attention to read- and write-modes, we consider modes to correspond to the privilege to perform particular **database** operations. We construct the lock mode compatibility matrix from the properties of the operations associated with the lock modes. Criteria for the "goodness" of a compatibility matrix, called correctness and maximal permissiveness, are defined and we show that our construction produces only correct, maximally permissive matrices. The construction is then extended to the case where transactions that hold multiple locks on an entity.

It is useful for reasons of avoiding or minimizing **deadlocks** to restrict the **types** of **lock** conversion that are allowed. These considerations motivated the introduction of update mode in IMS. We define a class called generalized update locks to describe all useful conversion privileges and prove bounds on the number of such modes.

Variable granularity (the ability of a transaction to lock data in

units ranging from small (e.g. records) to large (e.g. files or areas)) is usually implemented by means of intention locks. We extend the intention locks of System/R to produce a class of intention locks for any given set of lock modes, including generalized update locks.

The classes of locks we have introduced are used to define a deadlock-free, variable granularity locking protocol that, like the (deadlock-permitting) protocol of System/R, uses a directed acyclic graph of lockable entities to implement variable granularity. Deadlock-freedom is achieved by means of locks on the edges of the graph. The protocol is shown to be deadlock-free and to ensure the serializability of all legal schedules for an arbitrary set of lock modes.

*Work partially supported by NSF grant MCS-79-04528 and AFOSR grant AFOSR-80-0212.

16/5/26 (Item 1 from file: 2)

7101969 INSPEC Abstract Number: C2002-01-6160B-002

Title: Scalable session locking for a distributed file system

Author(s): Burns, R.C.; Rees, R.M.; Stockmeyer, L.J.; Long, D.D.E.
Author Affiliation: IBM Almaden Res. Center, San Jose, CA, USA

Journal: Cluster Computing vol.4, no.4 p.295-306
Publisher: Kluwer Academic Publishers,
Publication Date: 2001 Country of Publication: Netherlands
CODEN: CLCOFM ISSN: 1386-7857
SICI: 1386-7857(2001)4:4L.295:SSLD;1-Z
Material Identity Number: H401-2001-005
Language: English Document Type: Journal Paper (JP)
Treatment: Practical (P)

Abstract: **File** systems provide an interface for applications to obtain exclusive access to **files** , in which a process holds privileges to a **file** that cannot be preempted and restrict the capabilities of other processes. Local **file** systems do this by maintaining information about the privileges of current **file** sessions, and checking subsequent sessions for compatibility. Implementing exclusive access in this manner for distributed **file** systems degrades performance by requiring every new **file** session to be registered with a lock server that maintains the global session state. We present two techniques for improving the performance of session management in the distributed environment. We introduce a distributed lock for managing **file** access, called a semi-preemptible **lock** , that allows clients to **cache** privileges. Under a semi-preemptible **lock** , a **file** system creates new sessions without messages to the lock manager. This improves performance by exploiting locality - the affinity of **files** to clients. We also present data structures and algorithms for the dynamic evaluation of locks that allow a distributed **file** system to efficiently manage arbitrarily complex locking. In this case, complex means that an object can be locked in a large number of unique modes. The combination of these techniques results in a distributed locking scheme that supports fine-grained concurrency control with low memory and message overhead and with the assurance that their locking system is correct and avoids unnecessary **deadlocks** . (26 Refs)

Subfile: C
Descriptors: cache storage; concurrency control; data structures; distributed **databases** ; software performance evaluation
Identifiers: scalable session locking; distributed **file** system; exclusive access; process privileges; local **file** systems; current **file** sessions; compatibility checking; performance; lock server; global session state; session management; distributed lock; **file** access management; semi-preemptible lock; client-cached privileges; locality; **file** -client affinity; data structures; dynamic evaluation; locking modes; distributed locking scheme; fine-grained concurrency control; memory overhead; message overhead; **deadlocks** ; lock evaluation
Class Codes: C6160B (Distributed databases); C6120 (File organisation)

16/5/30 (Item 5 from file: 2)
DIALOG(R)File 2:INSPEC

5072600 INSPEC Abstract Number: C9511-6160B-022
Title: Concurrency control and recovery in multidatabase systems
Author(s): Pyung-Chul Kim; Won Kim; Yoon-Joon Lee
Author Affiliation: Korea Adv. Inst. of Sci. & Technol., Daejon, South Korea
Journal: Journal of Computer and Software Engineering vol.3, no.1 p.101-31
Publication Date: 1995 Country of Publication: USA
CODEN: JCOSE5 ISSN: 1069-5451
Language: English Document Type: Journal Paper (JP)
Treatment: Practical (P)
Abstract: A multidatabase system is a federation of independently built and administered component **database** systems which allows the users uniform access to more than one **database** within a transaction. One of the technical challenges in realizing a commercially viable multidatabase system is distributed transaction management that preserves the traditional serializability property of transactions even in the presence of system failures. We present a locking protocol and a recovery protocol for multidatabase systems. The locking protocol is an extension of the traditional two-phase locking protocol. We also introduce a **deadlock** detection technique based on the use of a potential-wait-for graph to detect **deadlocks** among multidatabase transactions being executed on the component **database** systems. Next, we present a commitment protocol, called a resubmit log method, for recovery from multidatabase transaction failures and system failures. The multidatabase two-phase locking protocol, the **deadlock** detection methods, and the resubmit **log** method of recovery can all be implemented without requiring any changes to existing **database** management systems. (18 Refs)
Subfile: C
Descriptors: access protocols; concurrency control; distributed **databases** ; system recovery; transaction processing
Identifiers: concurrency control; recovery protocol; multidatabase systems; federated **databases** ; uniform access; distributed transaction management; serializability; system failures; two-phase locking protocol; **deadlock** detection technique; potential-wait-for graph; commitment protocol; resubmit log method; transaction failures
Class Codes: C6160B (Distributed databases); C5640 (Protocols)

16/5/35 (Item 10 from file: 2)
DIALOG(R)File 2:INSPEC

04297397 INSPEC Abstract Number: C9301-6160-016
Title: Lock mode based resolution of uncompensatable deadlock **in compensating nested transactions**
Author(s): Takizawa, M.; Deen, S.M.
Author Affiliation: Dept. of Inf. & Syst. Eng., Tokyo Denki Univ., Saitama, Japan
Conference Title: Future Databases '92. Proceedings of the Second Far-East Workshop on Future Database Systems p.168-75
Editor(s): Qiming Chen; Yahiko Kambayashi; Sacks-Davis, R.
Publisher: World Scientific, Singapore
Publication Date: 1992 Country of Publication: Singapore xii+418 pp.
ISBN: 981 02 1040 X
Conference Date: 26-28 April 1992 Conference Location: Kyoto, Japan
Language: English Document Type: Conference Paper (PA)
Treatment: Practical (P)
Abstract: The authors discuss how to resolve **deadlock** in the interleaved execution of nested transactions. Since transactions in new applications like CAD require more objects for longer times than

conventional ones, there is higher possibility that **deadlock** occurs, and more **data** is **stored** in the log. In conventional **database** systems, when some **deadlock** occurs, one **deadlocked** transaction T is selected and the whole part is aborted by using the log which includes the old state. Another way to abort T is to execute a compensate operation op of each operation op in T. This method can reduce time for aborting and restarting T since only a part of T is aborted, and can reduce the log size since operations are stored in the log in stead of storing the state changed by the operations. The compensate operations may cause further **deadlock** , since the compensate operations require locks on the objects. They show that there exists uncompensatable **deadlock** which cannot be resolved by the compensate operations. Also, they show a method for resolving the uncompensatable **deadlock** by the compensate operations at the lowest level. (21 Refs)
Subfile: C
Descriptors: **database** management systems; system recovery; transaction processing
Identifiers: uncompensatable **deadlock** ; compensating nested transactions ; interleaved execution; CAD
Class Codes: C6160 (Database management systems (DBMS))

16/5/39 (Item 14 from file: 2)
DIALOG(R)File 2:INSPEC

03444445 INSPEC Abstract Number: C89055042
Title: Operation-specific locking in balanced structures
Author(s): Biliris, A.
Author Affiliation: Dept. of Comput. Sci., Boston Univ., MA, USA
Journal: Information Sciences vol.48, no.1 p.27-51
Publication Date: June 1989 Country of Publication: USA
CODEN: ISIJBC ISSN: 0020-0255
U.S. Copyright Clearance Center Code: 0020-0255/89/$03.50
Language: English Document Type: Journal Paper (JP)
Treatment: Theoretical (T)
Abstract: Balanced structures, variations of B-trees, have been used as an access aid for both primary and secondary indexing for quite some time. The paper presents a **deadlock** -free locking mechanism for B-trees in which different processes make use of different **lock types** in order to reach the leaf nodes. The compatibility relations among locks on a node do not exclusively depend on their type, but also on the node status and the number and kind of processes acting currently on the node. As a result, a number of insertion or deletion processes can operate concurrently on a node. The paper presents an appropriate recovery strategy in case of failure, and discusses the protocol modifications that are required so it can be used in other similar structures such as B/sup +/-trees, compressed B-trees, and R-trees for spatial searching. (38 Refs)
Subfile: C
Descriptors: concurrency control; **database** theory; search problems; trees (mathematics
Identifiers: primary indexing; balanced structures; B-trees; secondary indexing; deadlock-free; locking mechanism; insertion; deletion; recovery strategy; protocol modifications; spatial searching
Class Codes: C4250 (Database theory)

16/5/44 (Item 19 from file: 2)
DIALOG(R)File 2:INSPEC

01282737 INSPEC Abstract Number: C79002040
Title: Multiple-owner multiple- level deadlock **detection mechanism for resource contention control**
Author(s): Etchison, K.L.
Author Affiliation: IBM Corp., Armonk, NY, USA
Journal: IBM Technical Disclosure Bulletin vol.20, no.8 p.3066-9
Publication Date: Jan. 1978 Country of Publication: USA

CODEN: IBMTAA ISSN: 0018-8689
Language: English Document Type: Journal Paper (JP)
Treatment: Practical (P)
Abstract: A mechanism is described for use in multitask **data base** operations when multiple levels of access are allowed and a task may be waiting on more than one other task. (0 Refs)
Subfile: C
Descriptors: **database** management systems; hazards and race conditions; multi-access systems
Identifiers: deadlock detection mechanism; resource contention control; multitask **data base** operations; multiple levels of access
Class Codes: C6160 (Database management systems (DBMS))

16/5/45 (Item 20 from file: 2)
DIALOG(R)File 2:INSPEC

01017709 INSPEC Abstract Number: C77006926
Title: Integrity of data bases : **a general lockout algorithm with deadlock avoidance**
Author(s): Gardarin, G.; Spaccapietra, S.
Author Affiliation: Inst. de Programmation, Univ. Paris VI, Paris, France
Conference Title: Proceedings of the IFIP Working Conference on Modelling in Data Base Management Systems p.395-411
Editor(s): Nijssen, G.M.
Publisher: North-Holland, Amsterdam, Netherlands
Publication Date: 1976 Country of Publication: Netherlands vii+418 pp.
ISBN: 0 7204 0459 2
Conference Sponsor: IFIP
Conference Date: 5-8 Jan. 1976 Conference Location: Freudenstadt, West Germany
Language: English Document Type: Conference Paper (PA)
Treatment: Practical (P)
Abstract: Proposes elements for a theory on **database** concurrency, in which lockout rules are defined on 'objects' of any level and the concept of 'operation' is used as a generalization of the **type** -of-access notion. The **deadlock** problem is discussed. The authors present a detection algorithm, which uses a graph representing the waiting between users and looks for the existence of a circuit in it. Finally, using the external schema to anticipate possible operations on objects and Habermann's method to construct a deadlock avoidance algorithm starting from a detection algorithm, the authors propose a general method avoiding simultaneous execution of noncompatible operations on an object and avoiding deadlock situations to become effective. (9 Refs)
Subfile: C
Descriptors: **database** management systems; security of data
Identifiers: lockout algorithm; deadlock avoidance; **database** concurrency; detection algorithm; graph; **database** integrity
Class Codes: C6120 (File organisation)

16/5/46 (Item 21 from file: 2)
DIALOG(R)File 2:INSPEC

00537873 INSPEC Abstract Number: C73016887
Title: A practical approach to managing resources and avoiding deadlocks
Author(s): Frailey, D.J.
Author Affiliation: Southern Methodist Univ., Dallas, TX, USA
Journal: Communications of the ACM vol.16, no.5 p.323-9
Publication Date: May 1973 Country of Publication: USA
CODEN: CACMA2 ISSN: 0001-0782
Language: English Document Type: Journal Paper (JP)
Treatment: Practical (P)
Abstract: Resource scheduling and allocation can be expensive with regard to time and space in multiprogramming or time-sharing environments

involving large numbers of tasks and resources with conflicting requirements. Detection and/or prevention of deadlocks can require massive amounts of additional overhead if efficient usage of resources is to be maintained. A resource management program is described which uses linked lists along with other techniques to overcome a large portion of this overhead. The program, which is currently running as part of a large scale general purpose operating system, keeps resources relatively active but does not detect or prevent all **deadlocks** in its implemented state. Certain changes, which would permit more comprehensive **levels** of **deadlock** prevention/detection at additional cost, have not been incorporated in the running system due to the infrequency of **deadlock** situations. (10 Refs)
Subfile: C
Descriptors: **file** organisation; multiprogramming; operating systems (computers); storage allocation; time-sharing programs
Identifiers: resource scheduling; multiprogramming; deadlock; resource allocation; scheduling; multiprocessing; deadly embrace; time sharing
Class Codes: C6150J (Operating systems)

16/5/48 (Item 2 from file: 94)
DIALOG(R)File 94:JICST-EPlus

02557389 JICST ACCESSION NUMBER: 95A0762424 FILE SEGMENT: JICST-E
Parallel Transaction Management using Message Communication.
YOKOTA HARUO (1); ARAI RYUICHI (1)
(1) Hokurikusentankagakugijutsudaigakuindai
Joho Shori Gakkai Kenkyu Hokoku, 1995, VOL.95,NO.65(DBS-104), PAGE.105-112, FIG.3, REF.16
JOURNAL NUMBER: Z0031BAO ISSN NO: 0919-6072
UNIVERSAL DECIMAL CLASSIFICATION: 681.32 681.3:061.68
LANGUAGE: Japanese COUNTRY OF PUBLICATION: Japan
DOCUMENT TYPE: Journal
ARTICLE TYPE: Original paper
MEDIA TYPE: Printed Publication
ABSTRACT: We propose a model for an intelligent parallel transaction management system, and an implementation method of the system using a parallel logic programming language, KL1. Items in a **database** , nested transactions themselves, firing controls of active rules, and entries of a lock table are treated as sets of autonomous objects. We analyze messages among these objects, and mechanisms for detecting **deadlocks** and **logging** . KL1 is suitable for repressenting the system, since a perpetual process of KL1 directly corresponds to an object, and logical variables shared among the processes can be used for passing messages. (author abst.)
DESCRIPTORS: transaction processing; object-oriented **database** ; autonomous system; **deadlock** ; **DBMS** ; system model; logic programming language
BROADER DESCRIPTORS: treatment; **database** ; system; computer application system; model; programming language; formal language; language
CLASSIFICATION CODE(S): JC02010O; JD03030U

16/5/49 (Item 3 from file: 94)
DIALOG(R)File 94:JICST-EPlus

01756343 JICST ACCESSION NUMBER: 93A0514637 FILE SEGMENT: JICST-E
Resolution of Uncompensatable Deadlock .
YASUZAWA SHINJI (1); TAKIZAWA MAKOTO (1); DEEN S M (2)
(1) Tokyo Denki Univ., Faculty of Science and Engineering; (2) Univ.Keele
Joho Shori Gakkai Ronbunshi(Transactions of Information Processing Society of Japan), 1993, VOL.34,NO.5, PAGE.1045-1052, FIG.8, REF.25
JOURNAL NUMBER: Z0778AAZ ISSN NO: 0387-5806
UNIVERSAL DECIMAL CLASSIFICATION: 681.3:061.68
LANGUAGE: Japanese COUNTRY OF PUBLICATION: Japan
DOCUMENT TYPE: Journal
ARTICLE TYPE: Original paper

MEDIA TYPE: Printed Publication
ABSTRACT: In this paper, we discuss how to resolve **deadlock** occurred in interleaved execution of nested transactions. Since transactions in new applications like groupware systems and CAD require more objects for longer time than conventional ones, there is higher possibility that **deadlock** occurs, and more **data** has to be **stored** in the log. We discuss a method for resolving the **deadlock** where only a part of a **deadlocked** transaction T is aborted by executing the compensate operations. This method can reduce time for aborting and restarting T, and can reduce the log size. The compensate operations may cause further **deadlock** , since they require locks on the objects. We show that there exists uncompensatable **deadlock** which cannot be resolved by the compensate operations. Also, we show a method for resolving the uncompensatable **deadlock** by the compensate operations at the lowest level. (author abst.)
DESCRIPTORS: **database** ; **DBMS** ; transaction processing; **deadlock** ; concurrent control; abstract data type; data structure; reliability(property); parallel processing; distributed processing; compensation
BROADER DESCRIPTORS: computer application system; system; treatment; control; data type; mold and pattern; structure; performance
CLASSIFICATION CODE(S): JD03030U

16/5/51 (Item 5 from file: 94)
DIALOG(R)File 94:JICST-EPlus

01472238 JICST ACCESSION NUMBER: 91A0908606 FILE SEGMENT: JICST-E
Deadlock Resolution in Parallel Execution of Nested Transactions.
YASUZAWA S (1); TAKIZAWA M (1)
(1) Tokyo Denki Univ., Saitama, JPN
Tokyo Denki Daigaku Riko Gakubu Kiyo(Research Activities. Faculty of Science and Engineering of Tokyo Denki University), 1991, VOL.13, PAGE.45-52, FIG.6, REF.7
JOURNAL NUMBER: Z0885AAK ISSN NO: 0388-1989
UNIVERSAL DECIMAL CLASSIFICATION: 681.3:061.68
LANGUAGE: English COUNTRY OF PUBLICATION: Japan
DOCUMENT TYPE: Journal
ARTICLE TYPE: Original paper
MEDIA TYPE: Printed Publication
ABSTRACT: A distributed system is composed of multiple objects interconnected by communication networks. Each object is an abstract data type. Users write transactions to manipulate objects. Transactions are composed of operations and also atomic units of work for users. Each operation can call operations on another object. Suppose that one operation a on an object o calls two operations b on p and c on q. If b and c are independent, they can be called in parallel. This means that a can be executed in parallel. In this paper, we would like to discuss **deadlock** problems occurred when transactions are executed in parallel. In this paper, we define what **kind** of **deadlock** occurs when the transactions are concurrently executed. We present how to about the **deadlocked** transaction by executing the compensate operations. (author abst.)
DESCRIPTORS: **DBMS** ; distributed processing; parallel processing; transaction processing; deadlock; synchronous processing; data model; data structure; abstract data type; error detection
BROADER DESCRIPTORS: computer application system; system; treatment; model; structure; data type; mold and pattern; error control; control; detection
CLASSIFICATION CODE(S): JD03030U

16/5/53 (Item 7 from file: 94)
DIALOG(R)File 94:JICST-EPlus

00755054 JICST ACCESSION NUMBER: 89A0487666 FILE SEGMENT: JICST-E

Two step two phase locking mechanism adapted for real-time databases .
KAMBAYASHI YAHIKO (1); SAISHO KEIZO (1); ZHONG X (1)
(1) Kyushu Univ.
Joho Shori Gakkai Kenkyu Hokoku, 1989, VOL.89,NO.63(DBS-72), PAGE.131-138, FIG.9, REF.11
JOURNAL NUMBER: Z0031BAO ISSN NO: 0919-6072
UNIVERSAL DECIMAL CLASSIFICATION: 681.3:061.68
LANGUAGE: Japanese COUNTRY OF PUBLICATION: Japan
DOCUMENT TYPE: Journal
ARTICLE TYPE: Original paper
MEDIA TYPE: Printed Publication
ABSTRACT: In real-time **database** systems, transactions have deadline and priority and we must execute higher priority transactions earlier than lower priority transactions. In this paper, parallel transaction model, reservation of lock and dynamic exchange of reservation order are utilized in order to adapt two phase locking mechanism to real-time **databases** . Parallel transaction can reduce the number of transaction executing concurrently and reduce the possibility of **deadlocks** . The mechanism consists of two **levels** , reservation **level** and real **locking level** . At the reservation **level** priority can be reflected.(author abst.)
DESCRIPTORS: real time processing; **database** ; concurrent control; transaction processing; parallel processing; **DBMS** ; operating system; competitive problem; **database** machine; priority processing; deadlock; priority
BROADER DESCRIPTORS: treatment; control; computer application system; system; system program; computer program; software; problem; special purpose processor; hardware; right
CLASSIFICATION CODE(S): JD03030U

16/5/59 (Item 1 from file: 144)
DIALOG(R)File 144:Pascal
(c) 2004 INIST/CNRS. All rts. reserv.

10612298 PASCAL No.: 93-0121564
Adaptable concurrency control for atomic data types
ATKINS M S; COADY M Y
Simon Fraser univ., school computing sci., Burnaby BC V5A 1S6, Canada
Journal: ACM transactions on computer systems, 1992, 10 (3) 190-225
ISSN: 0734-2071 CODEN: ACSYEC Availability: INIST-21527;
354000031871100020
No. of Refs.: 35 ref.
Document Type: P (Serial) ; A (Analytic)
Country of Publication: USA
Language: English

English Descriptors: Performance evaluation; Synchronization; Mutual exclusion; **Deadlock** ; Abstract data **type** ; Shared memory; Concurrency control; Concurrent program; Distributed system; Distributed **database** ; Computer system

French Descriptors: Evaluation performance; Synchronisation; Exclusion mutuelle; Interblocage; Type abstrait; Memoire partagee; Controle concurrence; Programme concurrent; Systeme reparti; Base donnee repartie; Systeme informatique

Classification Codes: 001D02B04

16/5/65 (Item 4 from file: 95)
DIALOG(R)File 95:TEME-Technology & Management
(c) 2004 FIZ TECHNIK. All rts. reserv.

00503435 E91104512080
A recoverable and concurrent locking algorithm for a parallel semantic binary database machine with inverted replication
(Ein wiederaufsetzbarer und paralleler Blockierungsalgorithmus fuer einen

parallelen, semantischen, binaeren Parallelrechner mit in tierter Kopie)
Ghaly, R; Naphtali Rishe
Florida International Univ. Miami, USA
Intelligent Distributed Processing, Proceedings of the ISMM International Conference, Fort Lauderdale, USA, December 13-15, 19891990
Document type: Conference paper Language: English
Record type: Abstract
ISBN: 0-88986-138-2

ABSTRACT:
A recoverable and concurrent locking algorithm is proposed for the Linear-throughput Semantic **Database** Machine (LSDM), a multidisk, multi-processor **database** machine that offers massive parallelism. The **database** environment is based on the Semantic Binary Model (SBM), a fact-oriented representation of an informations system. The whole **database** is represented by a set of facts and their inverted replicas, which are distributed evenly with balanced segments among the available processors. Our algorithm employs a locking strategy based on time-stamp ordering of transactions. The requested locks depend on the granularity of data which is selected to be either a fact or an object (a contiguous range of facts associated with one entity). The algorithm is **deadlock** free since transactions are globally ordered among all sites. Fairness and freedom from starvation are ensured, since transactions are partially ordered over the waiting queues of a particular site. The recovery algorithm monitors and controls the execution of transactions so that the fact base includes only the results of the committed transactions.

File 347:JAPIO Nov 1976-2004/Mar(Updated 040708)
(c) 2004 JPO & JAPIO
File 350:Derwent WPIX 1963-2004/UD,UM &UP=200448F
(c) 2004 Thomson Derwent

Set	Items	Description
S1	1820	DEADLOCK??? OR DEAD()LOCK???
S2	12155	(S1 OR LOCK???)(5N)(LEVEL? ? OR TIER? ? OR GRADE? ? OR STRENGTH OR DEGREE? ? OR KIND? ? OR TYPE? ? OR CATEGOR? OR CLASS?? OR CLASSIFICATION OR FAMILY OR FAMILIES)
S3	1328	(S1 OR LOCK???)(5N)(HISTORY OR HISTORIES OR LOG???? OR CACHE? ? OR BUFFER? ?)
S4	189	S1(5N)(DATA OR INFORMATION OR NUMBER? ? OR AMOUNT? ? OR STATISTIC?? OR PROFILE? ?)
S5	19	S4(5N)(RETAIN??? OR STOR??? OR RECORD??? OR SAV??? OR MAINTAIN??? OR KEEP??? OR KEPT OR TRACK??? OR MONITOR???)
S6	155545	DATABASE? ? OR DATA()(BASE? ? OR WAREHOUSE? ?) OR DBMS OR RDBMS OR REPOSITOR??? OR (DATA OR INFORMATION)()MANAGEMENT
S7	2	S1 AND S2 AND (S3 OR S5) AND S6
S8	7	S1 AND S2 AND (S3 OR S5)
S9	7	S1 AND S2 AND S6
S10	7	S1 AND (S3 OR S5) AND S6
S11	81	S1 AND S2
S12	53	S1(5N)(LEVEL? ? OR TIER? ? OR GRADE? ? OR STRENGTH OR DEGREE? ? OR KIND? ? OR TYPE? ? OR CATEGOR? OR CLASS?? OR CLASSIFICATION OR FAMILY OR FAMILIES)
S13	26	S12 AND IC=G06F
S14	85	S1 AND (S3 OR S5)
S15	72	S14 AND IC=G06F
S16	66	S1 AND S3
S17	19	S1 AND S5
S18	700	(S1 OR LOCK???)(5N)(HISTORY OR HISTORIES OR LOG????)
S19	18	S1 AND S18
S20	64	S7:S10 OR S13 OR S17 OR S19
S21	17	S7:S10
S22	35	S13 OR S21
S23	16	S1(5N)(HISTORY OR HISTORIES OR LOG????)
S24	48	S22:S23
S25	45	S24 AND IC=G06F
S26	16	S25 AND AC=US/PR
S27	15	S26 AND AY=(1970:2002)/PR
S28	41	S25 AND PY=1970:2002
S29	4	S1(5N)READ???(5N)WRIT???

28/5/1 (Item 1 from file: 347)
DIALOG(R)File 347:JAPIO

06386494 **Image available**
MECHANISM FOR SOLVING REQUEST FOR GENERATING DEADLOCK OF COMPUTER SYSTEM

PUB. NO.: 11-328141 [JP 11328141 A]
PUBLISHED: November 30, 1999 (**19991130**)
INVENTOR(s): CHRISTIN C JONES
PAKU-KIN-MAKU
MICHAEL A BLAKE
MICHAEL FII
GARY E STRAIGHT
APPLICANT(s): INTERNATL BUSINESS MACH CORP <IBM>
APPL. NO.: 11-110569 [JP 99110569]
FILED: April 19, 1999 (19990419)
PRIORITY: 70432 [US 70432], US (United States of America), April 30, 1998 (19980430)
INTL CLASS: **G06F-015/177**

ABSTRACT

PROBLEM TO BE SOLVED: To dissolve a latent lockout between requests of different types competing for resources by blocking a **deadlock** between plural request **types** competing for access to the same resource.
SOLUTION: Internal hang detection logic 140 receives a subset 142 of global hang detection pulses from global hang pulse logic 141. The subset 142 is supplied to the 3-bit counter 204 of the internal hang detection logic 140, and the 3-bit counter 204 counts the global hang detection pulses until a count matches with a fixed value register and generates internal hang detection pulses at the point of time of matching. Also, generation of the internal hang detection pulses is inhibited at the time of setting the fixed value register to all 0 and the **deadlock** between the respective request **types** is blocked by disabling its use.

28/5/2 (Item 2 from file: 347)
DIALOG(R)File 347:JAPIO

06000143 **Image available**
DATA BASE MANAGEMENT SYSTEM

PUB. NO.: 10-283243 [JP 10283243 A]
PUBLISHED: October 23, 1998 (**19981023**)
INVENTOR(s): WATANABE MIKI
TANAKA KEI
HAYATA HIROSHI
APPLICANT(s): FUJI XEROX CO LTD [359761] (A Japanese Company or Corporation), JP (Japan)
APPL. NO.: 09-131690 [JP 97131690]
FILED: May 06, 1997 (19970506)
INTL CLASS: [6] **G06F-012/00** ; **G06F-012/00** ; **G06F-009/46**
JAPIO CLASS: 45.2 (INFORMATION PROCESSING -- Memory Units); 45.1 (INFORMATION PROCESSING -- Arithmetic Sequence Units)

ABSTRACT

PROBLEM TO BE SOLVED: To efficiently operate an application program in a multi-thread environment and to suppress an unnecessary wait time in the synchronous processing of execution right security by preventing the occurrence of **dead lock** between the control of security of an execution right and that of lock.

SOLUTION: After an execution right is secured in one of threads 11 to 13 by a synchronous processing part 22 with respect to a **cache** resource 15, **lock** of a transaction operated on the thread, on which the execution

right is secured, is secured by an exclusive control part 25 with respect to a resource 18 on a **data base** 17 corresponding to the resource 15. If lock of the transaction cannot be secured, the exclusive control part 25 reports a failure of lock to the synchronous processing part 22 to make it forcibly abandon the already secured execution right and allows another waiting thread to secure the execution right, thus avoiding the occurrence of **dead lock** . The synchronous processing related to the execution right is performed in the unit of data of the exclusive control processing related to lock, and plural synchronous processings are executed without bringing about the wait state with respect to data which have no dependence relations in the exclusive control.

28/5/4 (Item 4 from file: 347)
DIALOG(R)File 347:JAPIO

05780421 **Image available**
EXCLUSIVE CONTROL SYSTEM FOR RESOURCE

PUB. NO.: 10-063521 [JP 10063521 A]
PUBLISHED: March 06, 1998 (**19980306)**
INVENTOR(s): NISHIMURA OSAMU
APPLICANT(s): NEC SOFTWARE LTD [491061] (A Japanese Company or Corporation), JP (Japan)
APPL. NO.: 08-214544 [JP 96214544]
FILED: August 14, 1996 (19960814)
INTL CLASS: [6] **G06F-009/46**
JAPIO CLASS: 45.1 (INFORMATION PROCESSING -- Arithmetic Sequence Units)

ABSTRACT

PROBLEM TO BE SOLVED: To reduce the danger of a failure in securing resources for a program of high importance owing to a **deadlock** by making a **logical** resource control means reject not a resource securing request of high priority, but a request of low priority relating to the deadlock.

SOLUTION: This system consists of programs 201-203 which operate by exclusively securing various resources, the resources 301-303 that the respective programs use, and a logical resource control means 101 which logically controls requests to exclusively secure resources from the respective programs. When a program enters a waiting process for some resource, namely, when programs exclusively secure the resources, priority corresponding to importance is given to each program and if a deadlock is detected at the time of a resource securing request made by a program of high priority, a resource securing request made by a program of low priority having made the request to secure the relative resources is rejected

28/5/5 (Item 5 from file: 347)
DIALOG(R)File 347:JAPIO

04772335 **Image available**
PARALLEL PROCESSING SYSTEM USING ASSOCIATIVE MEMORY

PUB. NO.: 07-064935 [JP 7064935 A]
PUBLISHED: March 10, 1995 (**19950310)**
INVENTOR(s): KANEKO KATSUYUKI
APPLICANT(s): MATSUSHITA ELECTRIC IND CO LTD [000582] (A Japanese Company or Corporation), JP (Japan)
APPL. NO.: 05-212666 [JP 93212666]
FILED: August 27, 1993 (19930827)
INTL CLASS: [6] **G06F-015/173** ; **G06F-015/16**
JAPIO CLASS: 45.4 (INFORMATION PROCESSING -- Computer Applications); 45.1 (INFORMATION PROCESSING -- Arithmetic Sequence Units)

ABSTRACT

PURPOSE: To provide a constitution for realizing interprocessor transfer for which **deadlock** hardly occurs and the priority **degree** of communication is easily controlled.

CONSTITUTION: Respective element processors EP are mutually connected in a grid shape and the respective element processors 100 are constituted of a router 101, a communication buffer 102 and a processor 103. The communication buffer 102 is an associative memory and is constituted of a tag field 104 and a data field 105. The processor 103 supplies an associative key to the communication buffer 102 and the data corresponding to the key are obtained. For the transmission of the data, a tag and the data are written in the communication buffer 102 of a transmission destination. Reception is performed by retrieving the communication buffer 102 of its own with the remaining number of times of transfer stored in the tag and fetching the desired data. Thus, the respective element processors retrieve the communication buffers of their own with the remaining number of times of the transfer as the key and select the data to be transferred

28/5/6 (Item 6 from file: 347)
DIALOG(R)File 347:JAPIO

04693240 **Image available**
EXCLUSIVE CONTROL METHOD IN PARALLEL DATA ACCESS SYSTEM

PUB. NO.: 07-013840 [JP 7013840 A]
PUBLISHED: January 17, 1995 (**19950117)**
INVENTOR(s): YAMAHIRA KOSAKU
APPLICANT(s): HITACHI LTD [000510] (A Japanese Company or Corporation), JP (Japan)
APPL. NO.: 05-156907 [JP 93156907]
FILED: June 28, 1993 (19930628)
INTL CLASS: [6] **G06F-012/00**
JAPIO CLASS: 45.2 (INFORMATION PROCESSING -- Memory Units)

ABSTRACT

PURPOSE: To suppress a **deadlock** to nearly the same **degree** with processing by a single processor by deciding one access request as a wait or access error when both access requests are in a combination of update and update, or update and reference.

CONSTITUTION: Only when data storage areas 1a-1e to be accessed which are specified by access requests from two users overlap with one another at >=2 places as to data in one list format or file format, data to be accessed are locked before an access process request is sent to processors 3a-3e which access the data storage areas 1a-1e. Then one of them is decided as a wait or access error. Consequently, the number of the data areas 1a-1e that two users to >=2 of the data storage areas 1a-1e by the two users is no caused

28/5/8 (Item 8 from file: 347)
DIALOG(R)File 347:JAPIO

03892830 **Image available**
PREFERENTIAL PROGRAM SELECTING SYSTEM FOR **DEAD LOCK** GENERATION

PUB. NO.: 04-257930 [JP 4257930 A]
PUBLISHED: September 14, 1992 (**19920914)**
INVENTOR(s): KIKUCHI JUNICHI
APPLICANT(s): NIPPON TELEGR & TELEPH CORP <NTT> [000422] (A Japanese Company or Corporation), JP (Japan)
APPL. NO.: 03-020168 [JP 9120168]
FILED: February 13, 1991 (19910213)
INTL CLASS: [5] **G06F-009/46**
JAPIO CLASS: 45.1 (INFORMATION PROCESSING -- Arithmetic Sequence Units)

JOURNAL: Section: P, Section No. 1476, Vol. 17, No. [redacted] Pg. 35, January 28, 1993 (19930128)

ABSTRACT

PURPOSE: To reduce the rate of processing of applied program to be made invalid when generating a **dead lock** by invalidating the processing of the applied program of a low advance **degree** related to the **dead lock** up to the moment when the **dead lock** is generated.

CONSTITUTION: A referential program selection program 2 respectively calculates and holds the processing advance degree of an applied program 1 from the predictive value of the range of resources to be locked and the current value in advance as the selection reference of the applied program 1 to invalidate the processing up to the moment when the **dead lock** is generated. A **data base** management program 3 invalidates the processing of the applied program 1 of the low advance **degree** related to the **dead lock** up to the moment when the **dead lock** is generated. Then, the resources related to the **dead lock** are released, and the **dead lock** is canceled.

28/5/9 (Item 9 from file: 347)

DIALOG(R)File 347:JAPIO

03720532 **Image available**

DEADLOCK AVOIDING SYSTEM

PUB. NO.: 04-085632 [JP 4085632 A]
PUBLISHED: March 18, 1992 (**19920318)**
INVENTOR(s): ONO SHIGEYUKI
APPLICANT(s): NEC CORP [000423] (A Japanese Company or Corporation), JP (Japan)
APPL. NO.: 02-199008 [JP 90199008]
FILED: July 30, 1990 (19900730)
INTL CLASS: [5] **G06F-009/46** ; **G06F-009/46** ; **G06F-011/30** ; **G06F-012/00** ; **G06F-015/16**
JAPIO CLASS: 45.1 (INFORMATION PROCESSING -- Arithmetic Sequence Units); 45.2 (INFORMATION PROCESSING -- Memory Units); 45.4 (INFORMATION PROCESSING -- Computer Applications)
JOURNAL: Section: P, Section No. 1381, Vol. 16, No. 308, Pg. 135, July 07, 1992 (19920707)

ABSTRACT

PURPOSE: To prevent processes which are performed by a task of low execution level up to the occurrence of a deadlock from becoming useless by stopping and releasing a mask which is lower in execution **level** if the **deadlock** occurs to an SRR and resecuring the SRR at a time when both are started.

CONSTITUTION: The system is equipped with a periodic monitoring means 11 which decides the deadlock of the sequentially usable resources (SRR) periodically, a task detecting means 12 which detects a task in a deadlock state, and a batch releasing means 14 which release all SRRs locked by the low-execution- **level** task in the **deadlock** state and runs a task having a high execution **level** . If the **deadlock** occurs to an SRR, the task which is low in execution level is stopped temporarily and the SRR that it locks is released during the period; when the task is restarted, the SRR is resecured at a time. Consequently, the processes which are performed by the low-execution-level task up to the occurrence of the deadlock are prevented from becoming useless.

28/5/10 (Item 10 from file: 347)

DIALOG(R)File 347:JAPIO

03686235 **Image available**

DATA BASE PROCESSOR

PUB. NO.: 04-051335 [JP 4051335 A]
PUBLISHED: February 19, 1992 (**19920219)**
INVENTOR(s): KODERA MAKOTO
HIKITA SADAYUKI
APPLICANT(s): OKI ELECTRIC IND CO LTD [000029] (A Japanese Company or Corporation), JP (Japan)
APPL. NO.: 02-159685 [JP 90159685]
FILED: June 20, 1990 (19900620)
INTL CLASS: [5] **G06F-012/00 ; G06F-012/00**
JAPIO CLASS: 45.2 (INFORMATION PROCESSING -- Memory Units)
JOURNAL: Section: P, Section No. 1363, Vol. 16, No. 241, Pg. 28, June 03, 1992 (19920603)

ABSTRACT

PURPOSE: To recognize the developmental status of contention by providing a judging part which judges the possibility of occurrence of distributed **type dead lock** , and **monitoring** the **information** of transaction in which the contention occur in its own device by outputting to a device on a network.

CONSTITUTION: A notice part 15 informs the occurrence/dissipation of the contention to the judging part 16 by using a transaction ID. The judging part 16 recognizes a master transaction from the transaction ID of the transaction based on the information informed from the notice part 15, and compares the master transactions mutually, and judges the possibility of occurrence of the distributed **type dead lock** . A **dead lock** detector 17 inputs similar information from plural **data base** processors 11, and detects the occurrence of the distributed **type dead lock** on a system, and informs the judging part 16 of the occurrence. A simultaneous execution control part 12 receiving the notice from the judging part 16 instructs the release of standby processing (interruptive processing) to a transaction control part 11, and cancels the contention, thereby, the distributed **type dead lock** can be dissolved.

28/5/11 (Item 11 from file: 347)

03570838 **Image available**
MONITORING SYSTEM FOR PROGRAM RUNAWAY

PUB. NO.: 03-233738 [JP 3233738 A]
PUBLISHED: October 17, 1991 (**19911017)**
INVENTOR(s): YANO TSUNEJI
APPLICANT(s): FUJITSU LTD [000522] (A Japanese Company or Corporation), JP (Japan)
APPL. NO.: 02-030363 [JP 9030363]
FILED: February 09, 1990 (19900209)
INTL CLASS: [5] **G06F-011/30**
JAPIO CLASS: 45.1 (INFORMATION PROCESSING -- Arithmetic Sequence Units)
JOURNAL: Section: P, Section No. 1299, Vol. 16, No. 17, Pg. 48, January 16, 1992 (19920116)

ABSTRACT

PURPOSE: To detect the program abnormality of a task having a low **level** of priority despite its **deadlock** , etc., by setting the largest executing time for each task carried out by a CPU and monitoring the runaway in accordance with the processing time of each task.
CONSTITUTION: An internal OS of a memory 2-2 of a CPU 2-1 rewrites and sets the holding value of a latch 2-11 at the largest executing time via a latch buffer of a storage means 2-12 when the tasks 1-3 are started. Then the OS of the memory 2-2 reads out the count value of a counter 2-10 via a counter buffer of a storage means 2-13 at interruption of execution of a task and sets this count value to the latch 2-11 at execution of a task. Then the program runaway is monitored at the largest executing time set for each of tasks 1-3. Thus it is possible to detect the program abnormality of a task

having a low level of priority and to be processed by a CPU even through this task has an infinite loop or a deadlock, etc

28/5/15 (Item 15 from file: 347)
DIALOG(R)File 347:JAPIO

03133641 **Image available**
TRANSACTION CONTROL SYSTEM

PUB. NO.: 02-109141 [JP 2109141 A]
PUBLISHED: April 20, 1990 (**19900420)**
INVENTOR(s): ITO JUNKO
APPLICANT(s): NEC CORP [000423] (A Japanese Company or Corporation), JP (Japan)
APPL. NO.: 63-261406 [JP 88261406]
FILED: October 19, 1988 (19881019)
INTL CLASS: [5] **G06F-011/14** ; **G06F-015/00**
JAPIO CLASS: 45.1 (INFORMATION PROCESSING -- Arithmetic Sequence Units); 45.4 (INFORMATION PROCESSING -- Computer Applications)
JOURNAL: Section: P, Section No. 1076, Vol. 14, No. 329, Pg. 89, July 16, 1990 (19900716)

ABSTRACT

PURPOSE: To easily and surely carry out again the transactions by using the address following a stored transaction and the necessary information written into a data memory means and a log file every time the process is through with a transaction.

CONSTITUTION: The address which is executed after a transaction process program is written into a transaction start address memory area 5 as soon as a transaction process is through while a series of transactions are executed via a transaction execution control means 1. In the same way, all memory data stored in the transaction process program and a transaction end log are written into a memory data storage means 3 and a **log** file 11 respectively. When a **deadlock** is detected, a data restoring means 15 refers to the means 3 and the file 11 and resets a memory and a data file 9 contained in the process program to the states set before execution of the transaction. Therefore the transaction can be easily and surely executed again in accordance with the address of the area 5.

28/5/17 (Item 17 from file: 347)
DIALOG(R)File 347:JAPIO

02938750 **Image available**
DEADLOCK MANAGING SYSTEM

PUB. NO.: 01-236350 [JP 1236350 A]
PUBLISHED: September 21, 1989 (**19890921)**
INVENTOR(s): OIYAKE YASUKUNI
SEKI TOSHIBUMI
TAMURA SHINSUKE
APPLICANT(s): TOSHIBA CORP [000307] (A Japanese Company or Corporation), JP (Japan)
APPL. NO.: 63-063914 [JP 8863914]
FILED: March 17, 1988 (19880317)
INTL CLASS: [4] **G06F-015/16**
JAPIO CLASS: 45.4 (INFORMATION PROCESSING -- Computer Applications); 25.2 (MACHINE TOOLS -- Cutting & Grinding); 36.1 (LABOR SAVING DEVICES -- Industrial Robots)
JOURNAL: Section: P, Section No. 976, Vol. 13, No. 566, Pg. 59, December 15, 1989 (19891215)

ABSTRACT

PURPOSE: To efficiently manage two kinds of resources, one of which is

enabled to be reexecuted after discontinuation and the other is disabled to be reexecuted, by dividing a system into the two kinds of resources and notifying holding of the latter resource beforehand when the holding of the latter resource is requested.

CONSTITUTION: A system is divided into a computer 2 (Type 1 resource) whose reexecution after discontinuation is enabled and a group of a robot 4, lathe 5, and printer 6 (Type 2 resource) whose reexecution is disabled. Regarding the Type 2 resource, holding of a necessary resource group is notified beforehand before holding of the resources is requested and the request is permitted only when it is confirmed that occurrence of a **deadlock** is avoided. Regarding the **Type** 1 resource, on the other hand, holding of the resource is directly requested without making any advance notice. Therefore, the executing extent of a deadlock detecting means can be limited within the Type 1 resource and that of a preventing means can be limited within the Type 2 resource.

28/5/18 (Item 18 from file: 347)

02719533 **Image available**
EXCLUSIVE PROCESSING SYSTEM FOR COMPUTER **DATA BASE** SYSTEM

PUB. NO.: 01-017133 [JP 1017133 A]
PUBLISHED: January 20, 1989 (**19890120**)
INVENTOR(s): KOBARI KOUJI
APPLICANT(s): FUJI XEROX CO LTD [359761] (A Japanese Company or Corporation), JP (Japan)
APPL. NO.: 62-173048 [JP 87173048]
FILED: July 13, 1987 (19870713)
INTL CLASS: [4] **G06F-012/00** ; **G06F-007/22**
JAPIO CLASS: 45.2 (INFORMATION PROCESSING -- Memory Units); 45.1 (INFORMATION PROCESSING -- Arithmetic Sequence Units)
JOURNAL: Section: P, Section No. 867, Vol. 13, No. 191, Pg. 88, May 09, 1989 (19890509)

ABSTRACT

PURPOSE: To reduce the labor needed for the **deadlock** processing at the user side by checking the possibility of a **deadlock** when a file is opened and giving the open permission only when no possibility of the **deadlock** is confirmed even at the record operation.

CONSTITUTION: When a file opening request is received from a user and the relevant file is being used, it is decided whether the file can be shared or not. If so, the possibility of 'waiting' is checked. When the possibility of 'waiting' is confirmed, a **deadlock** is checked at the record **level** . If no **deadlock** is confirmed, a file opening process is carried out. While the occurrence of a **deadlock** is confirmed, the occurrence of this **deadlock** is informed to the user. At the same time, the **deadlock** check is carried out at both file and record levels when it is decided the sharing of a file is impossible. Then a waiting process is carried out if no **deadlock** occurs.

28/5/20 (Item 20 from file: 347)

02393351 **Image available**
CONTROL SYSTEM FOR RECORD INDICATOR

PUB. NO.: 63-010251 [JP 63010251 A]
PUBLISHED: January 16, 1988 (**19880116**)
INVENTOR(s): SHIMAZU KAZUYUKI
APPLICANT(s): NEC CORP [000423] (A Japanese Company or Corporation), JP (Japan)

APPL. NO.: 61-154903 [JP 86154903]
FILED: July 01, 1986 (19860701)
INTL CLASS: [4] **G06F-012/00 ; G06F-009/46**
JAPIO CLASS: 45.2 (INFORMATION PROCESSING -- Memory Units); 45.1 (INFORMATION PROCESSING -- Arithmetic Sequence Units)
JOURNAL: Section: P, Section No. 718, Vol. 12, No. 214, Pg. 42, June 18, 1988 (19880618)

ABSTRACT

PURPOSE: To reset the information on a **record** indicator when a **deadlock** occurs in a **data base** system by resetting said information saved when a retreat resetting instruction is executed and setting the record indicator back to the immediately preceding static point.

CONSTITUTION: A record indicator saving means 301 collects the information on the record indicator of a user program 1 at every time a commitment instruction is carried out and saves the collected information to a record indicator saving area 101. Then the control is shifted to a retreat restoring instruction executing means 4 after a retreat restoring instruction is carried out by the program 1. Thus the processing given to a **data base** file 2 from the immediately preceding static point is invalidated and a record indicator resetting means 401 is called out. The means 401 extracts the information on the record indicator out of the area 101 and restores it to the program 1. Thus the record indicator is fixed at the immediately preceding static point.

28/5/22 (Item 22 from file: 347)

01412350 **Image available**
PROGRAM CONTROLLER

PUB. NO.: 59-123950 [JP 59123950 A]
PUBLISHED: July 17, 1984 (**19840717)**
INVENTOR(s): UEDA KATSUHIKO
APPLICANT(s): MATSUSHITA ELECTRIC IND CO LTD [000582] (A Japanese Company or Corporation), JP (Japan)
APPL. NO.: 57-233602 [JP 82233602]
FILED: December 29, 1982 (19821229)
INTL CLASS: [3] **G06F-011/30 ; G06F-009/30 ; G06F-009/38**
JAPIO CLASS: 45.1 (INFORMATION PROCESSING -- Arithmetic Sequence Units)
JAPIO KEYWORD: R131 (INFORMATION PROCESSING -- Microcomputers & Microprocessers)
JOURNAL: Section: P, Section No. 314, Vol. 08, No. 252, Pg. 140, November 17, 1984 (19841117)

ABSTRACT

PURPOSE: To evade a deadlock state even in case of malfunction of a program counter or when an undefined instruction is inputted to an instruction register, by providing a deadlock detecting circuit.

CONSTITUTION: For example, the undefined instruction is inputted to the instruction register 5 owing to the malfunction of the program counter 1 or the destruction of the contents of a memory 2. In this case, decoding is not performed by a program lock array PLA6, so all outputs of the PLA6 have **logic** 0. Then, the **deadlock** state is entered. A state signal (d), however, generates logic 1 and a control signal (e) generates **logic** 0, so the **deadlock** detecting circuit 7 has **logic** 1. This is decoded by the PLA6 and a control signal (b) has logic 1 to load preset data set in the counter 1. This preset data is regarded as the starting address of a program to perform processing after the deadlock.

28/5/23 (Item 23 from file: 347)

00648323 **Image available**
MULTI-MICRO-PROCESSOR CONTROL SYSTEM

PUB. NO.: 55-135923 [JP 55135923 A]
PUBLISHED: October 23, 1980 (**19801023)**
INVENTOR(s): SHINOZAKI MASATSUGU
APPLICANT(s): HITACHI LTD [000510] (A Japanese Company or Corporation), JP (Japan)
APPL. NO.: 54-042662 [JP 7942662]
FILED: April 09, 1979 (19790409)
INTL CLASS: [3] **G06F-003/00** ; **G06F-015/16**
JAPIO CLASS: 45.3 (INFORMATION PROCESSING -- Input Output Units); 45.4 (INFORMATION PROCESSING -- Computer Applications)
JAPIO KEYWORD: R131 (INFORMATION PROCESSING -- Microcomputers & Microprocessers)
JOURNAL: Section: P, Section No. 44, Vol. 05, No. 5, Pg. 81, January 14, 1981 (19810114)

ABSTRACT

PURPOSE: To make it possible not to cause a **deadlock** state, by providing two different **kinds** of hardware for both the main system bus and the sub-system bus.

CONSTITUTION: In case the main system transfers a data to the sub-system, a main memory address data, a byte count data and a memory address data which are given by the main processor are taken into an address register 104, byte counters 102 and 103, and an address register 105, respectively. Afte that, a data is read out from the main system memory, is accumulated in a high speed buffer memory 106, and subsequently a data of the memory 106 is sent out to the sub-system. On the contrary, in case the sub-system transfers a data to the main system, an address data and a byte count data, which are given by the sub-processor are taken into the address registers 114 and 115, and the byte counters 112 and 113. Subsequently, a data of the memory of the sub-system is transferred to the main system through a high speed buffer memory 116

28/5/27 (Item 4 from file: 350)
DIALOG(R)File 350:Derwent WPIX

013881811 **Image available**
WPI Acc No: 2001-366023/ **200138**
XRPX Acc No: N01-266936
Executing variable delay system bus operations of differing type **or character without** deadlock **using shared** buffers **, involves indicating completion of operation after each cache has completed same operation**
Patent Assignee: INT BUSINESS MACHINES CORP (IBMC)
Inventor: ARIMILLI R K; KAISER J M; WILLIAMS D E
Number of Countries: 001 Number of Patents: 001
Patent Family:

Patent No	Kind	Date	Applicat No	Kind	Date	Week	
US 6202131	B1	20010313	US 984147	A	19980107	200138	B

Priority Applications (No Type Date): US 984147 A 19980107
Patent Details:

Patent No	Kind	Lan	Pg	Main IPC	Filing Notes
US 6202131	B1		14	G06F-013/00	

Abstract (Basic): US 6202131 B1

NOVELTY - The method involves detecting the presence of bus operations on a system bus (208). The initiation source and type of the detected operation, and the instance of a multiply initiated operation in each cache are then indicated. The detected operation in each cache is accepted at the same time. The completion of an operation is indicated only after each cache has completed the same operation.

DETAILED DESCRIPTION - The system bus can singly sourced and singly

initiated, or singly sourced and multiply initiated, multiply sourced and singly initiated, or multiply sourced and multiply initiated. An INDEPENDENT CLAIM is also included for the execution apparatus for executing variable delay system bus operations without **deadlock** using shared **buffers** .

USE - For executing variable delay system bus operations of differing **type** or character without **deadlock** using shared **buffers** .

ADVANTAGE - Prevents occurrence of **deadlock** from execution of variable delay system bus operations. Allows execution of variable time bus operations using shared **buffers** while avoiding ping-pong **deadlock** . Keeps any **cache** from finishing operation before any of the other caches.

DESCRIPTION OF DRAWING(S) - The figure is a block diagram illustrating a snooper in greater detail using the execution of variable delay system bus operations without **deadlock** .

System bus (208)

pp; 14 DwgNo 4/5

Title Terms: EXECUTE; VARIABLE; DELAY; SYSTEM; BUS; OPERATE; DIFFER; TYPE; CHARACTER; **DEADLOCK** ; SHARE; BUFFER; INDICATE; COMPLETE; OPERATE; AFTER; CACHE; COMPLETE; OPERATE

Derwent Class: T01

International Patent Class (Main): **G06F-013/00**

File Segment: EPI

28/5/28 (Item 5 from file: 350)

013881804 **Image available**

WPI Acc No: 2001-366016/ **200138**

XRPX Acc No: N01-266929

Bridge for computer systems, has arbiter to wait in response to stamp assertion, without granting lower priority requests to access pipe, until subsequent transaction from first device makes progress

Patent Assignee: INTEL CORP (ITLC)

Inventor: BOGIN Z; GADAGKAR A; KHANDEKAR N; LENT D D

Number of Countries: 001 Number of Patents: 001

Patent Family:

Patent No	Kind	Date	Applicat No	Kind	Date	Week	
US 6202112	B1	20010313	US 98205351	A	19981203	200138	B

Priority Applications (No Type Date): US 98205351 A 19981203

Patent Details:

Patent No	Kind	Lan	Pg	Main IPC	Filing Notes
US 6202112	B1		13	G06F-013/42	

Abstract (Basic): US 6202112 B1

NOVELTY - The bridge has an outbound pipe for buffering transaction information and data being transported from various devices to a bus. An arbiter (320) grants requests associated with the devices to access pipe, to transfer transaction information and data into the pipe.

DETAILED DESCRIPTION - If the pipe is unavailable to accept further transaction information or data, a reject signal is generated in response to an initial request associated with an initial transaction from a first one of the devices. A response control logic (316) generates a retry response for the initial transaction in response to the reject signal. A stamp signal is asserted in response to the reject signal. The arbiter in response to the stamp being asserted waits, without granting any other lower priority requests to access the pipe, until a subsequent transaction from the first device makes progress. An INDEPENDENT CLAIM is also included for a computer system.

USE - For computer systems.

ADVANTAGE - Avoids deadlocks without requiring expensive modifications in peripheral devices.

DESCRIPTION OF DRAWING(S) - The figures show a portion of a bridge having reject, stamp, and reject counter **logic** for avoiding **deadlock**

and livelock.
Response control logic (316)
Arbiter (320)
pp; 13 DwgNo 3/6
Title Terms: BRIDGE; COMPUTER; SYSTEM; ARBITER; WAIT; RESPOND; STAMP; LOWER ; PRIORITY; REQUEST; ACCESS; PIPE; SUBSEQUENT; TRANSACTION; FIRST; DEVICE ; PROGRESS
Derwent Class: T01
International Patent Class (Main): **G06F-013/42**
File Segment: EPI

28/5/31 (Item 8 from file: 350)
DIALOG(R)File 350:Derwent WPIX

012989511 **Image available**
WPI Acc No: 2000-161364/ **200014**
XRPX Acc No: N00-120365
Deadlock **risk handling method between simultaneous transaction in** database
Patent Assignee: TELEFONAKTIEBOLAGET ERICSSON L M (TELF)
Inventor: RONSTROEM M; RONSTROEM U M
Number of Countries: 087 Number of Patents: 010
Patent Family:

Patent No	Kind	Date	Applicat No	Kind	Date	Week	
WO 200005648	A1	20000203	WO 99SE1301	A	19990722	200014	B
SE 9802598	A	20000123	SE 982598	A	19980722	200019	
AU 9956602	A	20000214	AU 9956602	A	19990722	200029	
BR 9912294	A	20010417	BR 9912294	A	19990722	200128	
			WO 99SE1301	A	19990722		
EP 1097419	A1	20010509	EP 99943527	A	19990722	200128	
			WO 99SE1301	A	19990722		
US 6275823	B1	20010814	US 99358432	A	19990722	200148	
KR 2001071955	A	20010731	KR 2001700747	A	20010117	200208	
CN 1319203	A	20011024	CN 99811211	A	19990722	200213	
JP 2002521747	W	20020716	WO 99SE1301	A	19990722	200261	
			JP 2000561556	A	19990722		
SE 521433	C2	20031104	SE 982598	A	19980722	200374	

Priority Applications (No Type Date): SE 982598 A 19980722
Patent Details:

Patent No	Kind	Lan	Pg	Main IPC	Filing Notes
WO 200005648	A1	E	50	G06F-009/46	
Designated States (National): AE AL AM AT AU AZ BA BB BG BR BY CA CH CN CU CZ DE DK EE ES FI GB GD GE GH GM HR HU ID IL IN IS JP KE KG KP KR KZ LC LK LR LS LT LU LV MD MG MK MN MW MX NO NZ PL PT RO RU SD SE SG SI SK SL TJ TM TR TT UA UG UZ VN YU ZA ZW					
Designated States (Regional): AT BE CH CY DE DK EA ES FI FR GB GH GM GR IE IT KE LS LU MC MW NL OA PT SD SE SL SZ UG ZW					
SE 9802598	A			G06F-009/46	
AU 9956602	A			G06F-009/46	Based on patent WO 200005648
BR 9912294	A			G06F-009/46	Based on patent WO 200005648
EP 1097419	A1	E		G06F-009/46	Based on patent WO 200005648
Designated States (Regional): AT BE CH CY DE DK ES FI FR GB GR IE IT LI LU MC NL PT SE					
US 6275823	B1			G06F-017/30	
KR 2001071955	A			G06F-009/46	
CN 1319203	A			G06F-009/46	
JP 2002521747	W		51	G06F-012/00	Based on patent WO 200005648
SE 521433	C2			G06F-009/46	

Abstract (Basic): WO 200005648 A1
NOVELTY - A lock out queue (C1,C2) is allocated to respective objects in **database** . Transactions (T1-T8) that have requested access to an object are placed in the queue to await the access of a preceding transaction to the object. The transactions in the lock out queues are perused, to handle **deadlock** risk between transactions request access

to same objects.

DETAILED DESCRIPTION - The active or ongoing transactions in the **database** are divided into locking and non-locking combinations. The locking combination results in a **deadlock** . The locking transactions are compared with those affiliated to lockout queues, to establish whether or not the combination of transactions constitute a locking or non locking combination. The transactions are divided into simple and complex transactions. The simple transactions are those that access only one object in the **database** whereas complex transactions constitute the locking transactions and access two or more objects. The locking combination comprises complex transactions. The non-locking combination comprises simple transaction and simple or complex transaction. The transactions are also divided into writing and reading transactions. The locking combination comprises a writing transaction and reading or writing transaction. Non-loading combination comprises two reading transactions.

USE - For handling **deadlock** risk between simultaneous transactions in **database** .

ADVANTAGE - Reduces considerably the execution capacity required for monitoring the risk of **deadlock** relative to the capacity, while maintaining security against **deadlock** between simultaneous transactions in a **database** . Enable distinguishing locking and non **locking** combinations, readily by observing the **category** to which the concerned transactions belong. Transactions that are liable to cause **deadlocks** are easily identified.

DESCRIPTION OF DRAWING(S) - The figure illustrates two objects with associated lockout queues.

Queues (C1,C2)

Transactions (T1-T8)

pp; 50 DwgNo 2/19

Title Terms: **DEADLOCK** ; RISK; HANDLE; METHOD; SIMULTANEOUS; TRANSACTION; **DATABASE**

Derwent Class: T01

International Patent Class (Main): **G06F-009/46** ; **G06F-012/00** ; **G06F-017/30**

International Patent Class (Additional): **G06F-017/30**

File Segment: EPI

28/5/32 (Item 9 from file: 350)

DIALOG(R)File 350:Derwent WPIX

012192939 **Image available**

WPI Acc No: 1998-609852/ **199851**

XRPX Acc No: N98-474441

Parallel lock management method in multi-tasking database **system - involves protecting access to lock management data structures themselves with various locks containing spin locks**

Patent Assignee: SYBASE INC (SYBA-N)

Inventor: KRISHNAN V B; SANKARAN M

Number of Countries: 001 Number of Patents: 001

Patent Family:

Patent No	Kind	Date	Applicat No	Kind	Date	Week	
US 5832484	A	19981103	US 96673893	A	19960702	199851	B

Priority Applications (No Type Date): US 96673893 A 19960702

Patent Details:

Patent No	Kind	Lan	Pg	Main IPC	Filing Notes
US 5832484	A		52	G06F-017/00	

Abstract (Basic): US 5832484 A

The method involves providing each **database** with a lock manager for controlling access to objects in a **database** system. The lock management data structures which are shared among all **databases** are stored, that contains locking information about various **locks** comprising multiple **lock types** for protecting access to objects in a **database** system which are shared. The access to lock management

data structures are controlled through atleast one hash table (302,304,306) comprising multiple hash buckets.

Each hash bucket is associated with a particular **lock type** from various **lock types** provided in the system. The access to lock management data structures is protected themselves with another lock containing spin locks (312,314,316). Parallel access is performed to locks provided by **database** system, by associating each spin lock from locks with particular group of hash buckets.

USE - For controlling allocation of locks on information stored in **database** management system.

ADVANTAGE - Detects **deadlocks** among sleeping tasks effectively. Improves system performance.

Dwg.3A/9

Title Terms: PARALLEL; LOCK; MANAGEMENT; METHOD; MULTI; **DATABASE** ; SYSTEM; PROTECT; ACCESS; LOCK; MANAGEMENT; DATA; STRUCTURE; VARIOUS; LOCK; CONTAIN; SPIN; LOCK
Derwent Class: T01
International Patent Class (Main): **G06F-017/00**
File Segment: EPI

28/5/34 (Item 11 from file: 350)
DIALOG(R)File 350:Derwent WPIX

011890785 **Image available**
WPI Acc No: 1998-307695/ **199827**
XRPX Acc No: N98-241863

Parallel execution control system for communication apparatus, computer - includes manager provided with memory that stores lock information for management object, based on which parallel execution control of both manager and agent is carried out

Patent Assignee: FUJITSU LTD (FUIT)
Number of Countries: 001 Number of Patents: 001
Patent Family:

Patent No	Kind	Date	Applicat No	Kind	Date	Week	
JP 10111812	A	19980428	JP 9735245	A	19970219	199827	B

Priority Applications (No Type Date): JP 96215759 A 19960815
Patent Details:

Patent No	Kind	Lan	Pg	Main IPC	Filing Notes
JP 10111812	A		17	G06F-009/46	

Abstract (Basic): JP 10111812 A

The system includes a manager (MNG) and an agent (AGT). A lock information for management object of parallel execution control is stored in a memory of both manager and agent. Parallel execution controllers (2,6) carry out parallel execution control of both manager and agent.

When a dead-lock detecting unit in the manager carries out transaction with an instance lock, an internal starting of a dead-lock detection process is done, immediately after generation of instance lock command. When **class** lock command is generated, the **dead - lock** detection process is not started. A roll back indicator carries out roll back of the transactions which created dead-lock, when dead-lock is detected.

ADVANTAGE - Prevents degradation of capability of system. Facilitates increasing command response, effectively.

Dwg.1/9

Title Terms: PARALLEL; EXECUTE; CONTROL; SYSTEM; COMMUNICATE; APPARATUS; COMPUTER; MANAGE; MEMORY; STORAGE; LOCK; INFORMATION; MANAGEMENT; OBJECT; BASED; PARALLEL; EXECUTE; CONTROL; MANAGE; AGENT; CARRY
Derwent Class: T01
International Patent Class (Main): **G06F-009/46**
International Patent Class (Additional): **G06F-015/16**
File Segment: EPI

28/5/38 (Item 15 from file: 350)
DIALOG(R)File 350:Derwent WPIX

009271788 **Image available**
WPI Acc No: 1992-399200/ **199248**
XRPX Acc No: N92-304524
Routing in hierarchical inter-processor communications SIMD network - transmitting packets between large number of parallel processors using succession of routing cycles, preventing deadlock **between** levels **and** buffer **overflow**
Patent Assignee: SARNOFF RES CENT INC DAVID (SARN-N)
Inventor: CHIN D; LEE S
Number of Countries: 016 Number of Patents: 005
Patent Family:

Patent No	Kind	Date	Applicat No	Kind	Date	Week	
WO 9220177	A1	19921112	WO 92US2668	A	19920409	199248	B
US 5224100	A	19930629	US 91697556	A	19910509	199327	
EP 583400	A1	19940223	EP 92912852	A	19920409	199408	
			WO 92US2668	A	19920409		
JP 6507744	W	19940901	JP 92511818	A	19920409	199439	
			WO 92US2668	A	19920409		
EP 583400	A4	19950208	EP 92912852	A	19920000	199610	

Priority Applications (No Type Date): US 91697556 A 19910509
Cited Patents: US 4740954; US 5018133; US 5088091; US 5109379; No-Citns.
Patent Details:

Patent No	Kind	Lan	Pg	Main IPC	Filing Notes
WO 9220177	A1	E	9	H04J-003/24	
Designated States (National): JP					
Designated States (Regional): AT BE CH DE DK ES FR GB GR IT LU MC NL SE					
US 5224100	A		10	H04J-003/24	
EP 583400	A1	E		H04J-003/24	Based on patent WO 9220177
Designated States (Regional): DE FR GB IT NL					
JP 6507744	W			G06F-015/80	Based on patent WO 9220177
EP 583400	A4			H04J-003/24	

Abstract (Basic): WO 9220177 A
The process for a single-instruction-multiple-data (SIMD) multi-level hierarchical network of nodes (N111-N113, N121-N123 etc.) arranged in clusters and connected by dual, unidirectional channels includes a large number of parallel processors eg. 4096 arranged in a B cabinets each with B circuit boards each having 64 processors.
Packets including routing address information are sent during a succession of routing cycles from transmitting to receiving processors. Each node includes a storage buffer having a capacity of one more than the number of channels at that level.
ADVANTAGE - Prevents **deadlock** between **levels** and **buffer** overflow. Reduces hardware complexity by almost eliminating number of flow control lines. Minimises total routing latency time by maximising number of pockets present in network at any time.
Dwg.3/3
Title Terms: ROUTE; HIERARCHY; INTER; PROCESSOR; COMMUNICATE; SIMD; NETWORK ; TRANSMIT; PACKET; NUMBER; PARALLEL; PROCESSOR; SUCCESSION; ROUTE; CYCLE ; PREVENT; **DEADLOCK** ; LEVEL; BUFFER; OVERFLOW
Derwent Class: T01; W01
International Patent Class (Main): **G06F-015/80** ; H04J-003/24
International Patent Class (Additional): **G06F-015/16**
File Segment: EPI

28/5/39 (Item 16 from file: 350)
DIALOG(R)File 350:Derwent WPIX

009197694 **Image available**
WPI Acc No: 1992-325126/ **199240**
Related WPI Acc No: 1992-325064; 1992-325125; 1992-325127; 1992-325128;

1992-325445; 1992-3254[redacted]; 1993-328243; 1994-093743; 199[redacted]93747;
1994-093749; 1994-093789; 1994-094295; 1994-366262; 1995-008875;
1995-163664; 1995-292815; 1997-525950; 1998-230141

XRPX Acc No: N92-248541

Priority broadcast and multi-cast for un-buffered multi-stage network - permits multiple broadcasts to be queued at individual switching appts. which resolves broadcast contention on synchronous priority driven basis

Patent Assignee: INT BUSINESS MACHINES CORP (IBMC)

Inventor: BARKER T N; KOGGE P M; OLNOWICH H T; VANDLING G C

Number of Countries: 003 Number of Patents: 002

Patent Family:

Patent No	Kind	Date	Applicat No	Kind	Date	Week	
EP 505780	A2	19920930	EP 92103747	A	19920305	199240	B
EP 505780	A3	19931103				199511	

Priority Applications (No Type Date): US 91799262 A 19911127; US 91677543 A 19910329

Cited Patents: No-SR.Pub; 00 2.Jnl.Re; 00 WO0870; 30 AWO0870

Patent Details:

Patent No	Kind	Lan	Pg	Main IPC	Filing Notes
EP 505780	A2	E	40	G06F-015/16	

Designated States (Regional): DE FR GB

Abstract (Basic): EP 505780 A

The system has two modes using different priority levels assigned to each function and transmitted over the same physical single network path. A broadcast or multi-cast function is provided over the second mode of the switch. The broadcast apparatus comprises assignment means which assigns multiple priorities and holds various priorities pending in the switch until they are executed in a praioritised order.

When contention or connection blockages occur they are serviced based on a synchronised priority system where pending broadcast and multi-cast operations will be held pending at the switching system and executed as soon as the contention or blockage is removed. The hardware circuitry detects all the different **types** of **deadlock** conditions automatically and issues correction indications to the network paths involved.

ADVANTAGE - Permits multiple multi-cast operations to occur simultaneously within network. Has faster high powered broadcast and multi-cast function than prior art.

File 348:EUROPEAN PATENTS 1978-2004/Jul W03
(c) 2004 European Patent Office
File 349:PCT FULLTEXT 1979-2002/UB=20040729,UT=20040722
(c) 2004 WIPO/Univentio

Set	Items	Description
S1	1595	DEADLOCK??? OR DEAD()LOCK???
S2	15500	(S1 OR LOCK???)(5N)(LEVEL? ? OR TIER? ? OR GRADE? ? OR STRENGTH OR DEGREE? ? OR KIND? ? OR TYPE? ? OR CATEGOR? OR CLASS?? OR CLASSIFICATION OR FAMILY OR FAMILIES)
S3	718	(S1 OR LOCK???)(5N)(HISTORY OR HISTORIES OR LOG? ? OR LOGG???? OR CACHE? ?)
S4	211	S1(5N)(DATA OR INFORMATION OR NUMBER? ? OR AMOUNT? ? OR STATISTIC?? OR PROFILE? ?)
S5	19	S4(5N)(RETAIN??? OR STOR??? OR RECORD??? OR SAV??? OR MAINTAIN??? OR KEEP??? OR KEPT OR TRACK??? OR MONITOR???)
S6	146659	DATABASE? ? OR DATA()(BASE? ? OR WAREHOUSE? ?) OR DBMS OR RDBMS OR REPOSITOR??? OR (DATA OR INFORMATION)()MANAGEMENT
S7	2	S1(50N)S2(50N)(S3 OR S5)(50N)S6
S8	3	S1(50N)S2(50N)(S3 OR S5)
S9	32	S1(50N)S2(50N)S6
S10	8	S1(50N)(S3 OR S5)(50N)S6
S11	22	S1(50N)S2(50N)FILE? ?
S12	5	S1(50N)(S3 OR S5)(50N)FILE? ?
S13	51	S7:S12
S14	49	S13 AND IC=G06F
S15	2	S1(10N)GRANULAR?

14/3,K/1 (Item 1 from file: 348)
DIALOG(R)File 348:EUROPEAN PATENTS

01743426
Database constraint enforcer
Modul zur Erzwingung von Korrektheitsbedingungen in einer Datenbank
Module pour imposer des contraintes dans une base de donnees
PATENT ASSIGNEE:
IT Liberator AS, (4646830), General Birchs gate 16, 0454 Oslo, (NO),
(Applicant designated States: all)
INVENTOR:
Sjogren, Bjorn-Harald, Bekkestien 26B, 1406 Ski, (NO)
Bjornemyr, Jan-Thore, Kyhns gate 14, 2317 Hamar, (NO)
LEGAL REPRESENTATIVE:
Briddes, Sam et al (28656), Onsagers Ltd c/o Innovation Norway 5 Lower Regent Street, London SW1Y 4LR, (GB)
PATENT (CC, No, Kind, Date): EP 1426884 A1 040609 (Basic)
APPLICATION (CC, No, Date): EP 2003257444 031126;
PRIORITY (CC, No, Date): US 428937 P 021126
DESIGNATED STATES: AT; BE; BG; CH; CY; CZ; DE; DK; EE; ES; FI; FR; GB; GR; HU; IE; IT; LI; LU; MC; NL; PT; RO; SE; SI; SK; TR
EXTENDED DESIGNATED STATES: AL; LT; LV; MK
INTERNATIONAL PATENT CLASS: **G06F-017/30**
ABSTRACT WORD COUNT: 187
NOTE:
Figure number on first page: 4

LANGUAGE (Publication,Procedural,Application): English; English; English
FULLTEXT AVAILABILITY:

Available Text	Language	Update	Word Count
CLAIMS A	(English)	200424	754
SPEC A	(English)	200424	4719
Total word count - document A			5473
Total word count - document B			0
Total word count - documents A + B			5473

INTERNATIONAL PATENT CLASS: **G06F-017/30**

...SPECIFICATION to Divorced can only be checked when the status is changed. The present invention mainly relates to constraint enforcers for static rules.
To keep a **database** consistent at all times, sometimes needs very complex programming.
Some constraints are impossible to implement if they have to be checked per DML statement. One...
...and the insert is rejected. If you insert T2 first, the value does not exist in T1 and the insert is rejected. It is a **deadlock** situation.
For these **kinds** of problems, the term Conceptual Transaction has been introduced. It states that at the beginning and end of the transaction, the **database** must be in a consistent state. During the transaction the **database** is allowed to be in an inconsistent state. By using a Conceptual Transaction, the above examples become quite trivial.
A **Database** Transaction is a sequence of DML statements needed for a program to do a certain task. It may be thought of as an envelope with...

14/3,K/2 (Item 2 from file: 348)
DIALOG(R)File 348:EUROPEAN PATENTS

01616634
Algorithm for dynamic selection of data locking granularity
Algorithmus fur dynamische Auswahl von der Datenverriegelungsgranularitat
Algorithme pour la selection dynamique de la granularite de verrouillage de donnees
PATENT ASSIGNEE:
Openwave Systems Inc., (3397262), 1400 Seaport Boulevard, Office no.

W4152, Redwood City, CA 94063, (US), (Applicant designated States: all)
INVENTOR:
Holmgren, Steve, 117 W. Mason Street, Santa Barbara, CA 93101, (US)
LEGAL REPRESENTATIVE:
McLeish, Nicholas Alistair Maxwell et al (74621), Boult Wade Tennant
Verulam Gardens 70 Gray's Inn Road, London WC1X 8BT, (GB)
PATENT (CC, No, Kind, Date): EP 1335305 A2 030813 (Basic)
APPLICATION (CC, No, Date): EP 2002258180 021127;
PRIORITY (CC, No, Date): US 58164 020125
DESIGNATED STATES: AT; BE; BG; CH; CY; CZ; DE; DK; EE; ES; FI; FR; GB; GR;
IE; IT; LI; LU; MC; NL; PT; SE; SK; TR
EXTENDED DESIGNATED STATES: AL; LT; LV; MK; RO; SI
INTERNATIONAL PATENT CLASS: **G06F-017/30**
ABSTRACT WORD COUNT: 61
NOTE:
Figure number on first page: 1

LANGUAGE (Publication,Procedural,Application): English; English; English
FULLTEXT AVAILABILITY:

Available Text	Language	Update	Word Count
CLAIMS A	(English)	200333	1032
SPEC A	(English)	200333	2876
Total word count - document A			3908
Total word count - document B			0
Total word count - documents A + B			3908

INTERNATIONAL PATENT CLASS: **G06F-017/30**

...ABSTRACT A2

A method and apparatus for improving **database** concurrency are described. Upon receiving a request to access data, a data locality within a **database** may be determined utilizing unique data keys. An access to the data may be provided while locking part of the **database** based on a data **locking level** determined based on **deadlock history** corresponding to the data locality.

...SPECIFICATION A2

FIELD OF THE INVENTION

The present invention pertains to database technology. More particularly, the present invention relates to improving **database** concurrency while minimizing a possibility of a deadlock.

BACKGROUND OF THE INVENTION

Present technology allows multiple users to access one set of data via a...

...during the processing of a message. If the granularity of locks within a message system is too fine grained, this leads to deadlock situations.

A **deadlock** is a case where one thread of processing holds a lock and, at the same time, requires a lock held by another thread. In addition...

...the lock that the first thread holds. Without external intervention, this is an unresolvable situation where the processing of neither thread can progress. In addition, **deadlock** detection and external intervention slows message processing considerably, thus making the **deadlock** detection process inefficient.

A goal of the idealized message processing then is to minimize **deadlock** conditions, while allowing many threads of processing to access message data or message data infrastructure at the same time, thus improving data concurrency.

SUMMARY OF...

...described. The method may comprise receiving a request to access data, determining a data locality within a database utilizing unique data keys, determining a data **locking level** based on a **deadlock history** corresponding to the data locality, and providing access to the data while locking part of the database based on the data **locking level** .

history to the predetermined **deadlock** thresholds.

28. The apparatus of claim 21 wherein the unique data keys comprise user identification keys.

14/3,K/4 (Item 4 from file: 348)

DIALOG(R)File 348:EUROPEAN PATENTS

00826645

Transaction device driver technique for a journaling file system

Transaktionsgerattreiberverfahren fur ein Dateiensystem mit Logging-Moglichkeit

Procede de commande d'un dispositif de transaction pour un systeme de fichiers avec possibilite de logging

PATENT ASSIGNEE:
SUN MICROSYSTEMS, INC., (1392735), 2550 Garcia Avenue, MS PAL1-521, Mountain View, California 94043-1100, (US), (applicant designated states: DE;FR;GB;IT;NL)

INVENTOR:
Fuller, Billy J., 8510 Sutterfield Drive, Colorado Springs, Colorado 80920, (US)

LEGAL REPRESENTATIVE:
Hanna, Peter William Derek et al (72341), Tomkins & Co., 5 Dartmouth Road , Dublin 6, (IE)

PATENT (CC, No, Kind, Date): EP 767435 A1 970409 (Basic)
EP 767435 B1 990324
APPLICATION (CC, No, Date): EP 96650038 960904;
PRIORITY (CC, No, Date): US 526380 950911
DESIGNATED STATES: DE; FR; GB; IT; NL
INTERNATIONAL PATENT CLASS: **G06F-017/30** ; **G06F-011/14**
ABSTRACT WORD COUNT: 147

LANGUAGE (Publication,Procedural,Application): English; English; English
FULLTEXT AVAILABILITY:

Available Text	Language	Update	Word Count
CLAIMS B	(English)	9911	678
CLAIMS B	(German)	9911	571
CLAIMS B	(French)	9911	780
SPEC B	(English)	9911	8793
Total word count - document A			0
Total word count - document B			10822
Total word count - documents A + B			10822

INTERNATIONAL PATENT CLASS: **G06F-017/30** ...

... **G06F-011/14**

...SPECIFICATION 5% of ufs(underscore)ninode.

Some new counters may be added to inode stats structure:

ufs(underscore)iinactive frees the ondisk resources held by deleted **files** . Freeing inodes in ufs(underscore)iinactive () can **deadlock** be system as above-described and the same solution may be used, that is, deleted **files** are processed by a thread. The thread's queue is limited to ufs(underscore)ninode entries. ufs(underscore)rmdir() and ufs(underscore)remove() enforce the limit.

The system **deadlocks** if a thread holds the inode **cache** 's **lock** when it is suspended while entering a transaction. A thread suspends entering a transaction if there isn't sufficient log space at that time. The...

...functions ufs(underscore)flushi, ufs(underscore)iflush, and ufs(underscore)flush inodes use a single scan-inode-hash function that doesn't hold the inode **cache** **lock** :

ufs(underscore)iget uses the same protocol. This protocol is possible because the new iget/iinactive protocol obviates the problems inherent in attempting to reuse a **cached** inode.

The **lockfs** flush routine, ufs(underscore)flush inodes, is altered to

effectuate the present invention. ufs(underscore)flush-inodes hides inodes while flushing them. The inodes are...

14/3,K/6 (Item 6 from file: 348)
DIALOG(R)File 348:EUROPEAN PATENTS
(c) 2004 European Patent Office. All rts. reserv.

00662792
Process monitoring in a multiprocessing server
Prozessuberwachung in einem Mehrfachverarbeitungsanbieter
Surveillance de processus dans un serveur a multitraitement
PATENT ASSIGNEE:
International Business Machines Corporation, (200120), Old Orchard Road, Armonk, N.Y. 10504, (US), (applicant designated states: DE;FR;GB)
INVENTOR:
Chang, David Yu, 7004 Windridge Cove, Austin, Texas 78759, (US)
LEGAL REPRESENTATIVE:
Litherland, David Peter (75471), IBM United Kingdom Limited Intellectual Property Department Hursley Park, Winchester, Hampshire SO21 2JN, (GB)
PATENT (CC, No, Kind, Date): EP 636985 A1 950201 (Basic)
EP 636985 B1 980408
APPLICATION (CC, No, Date): EP 94305231 940715;
PRIORITY (CC, No, Date): US 97506 930727
DESIGNATED STATES: DE; FR; GB
INTERNATIONAL PATENT CLASS: **G06F-011/30**
ABSTRACT WORD COUNT: 194

LANGUAGE (Publication,Procedural,Application): English; English; English
FULLTEXT AVAILABILITY:

Available Text	Language	Update	Word Count
CLAIMS B	(English)	9815	409
CLAIMS B	(German)	9815	402
CLAIMS B	(French)	9815	531
SPEC B	(English)	9815	3568
Total word count - document A			0
Total word count - document B			4910
Total word count - documents A + B			4910

INTERNATIONAL PATENT CLASS: **G06F-011/30**

...SPECIFICATION are as follows:
- in free agent queue
- waiting on the database manager queue
- waiting on the parallel database queue
- processing database manager requests
- processing parallel **database** requests
- waiting on buffer queue services connection: token = xxxx, sid = x,
- waiting on closing buffer queue services connection: token = xxxx
- waiting on buffer distribution services...
...x, rid = x
- waiting on buffer queue services data: token = xxxx, sid = x, rid = x
- waiting on fast communication manager memory request
- waiting on parallel **database** agent shared information
- waiting on table access: table token = xxxx
- waiting on access to **data management** services **database** control block
- waiting on access to data protection services **database** control block

- waiting to access to data protection services read buffer
- waiting to write a **log**
- **deadlock** detector waiting for time out
- waiting for log I/O done

From the examples of the states identified above it becomes apparent that the server...

...proc-type: by the nature of the server process, the server processes can be grouped into different process types. A server process can be a

database agent process, communication process, a **deadlock** detector, et cetera. New process **types** can be created by the applications.

proc(underscore)state: a process is either in "runnable" or "waiting" state.

syn(underscore)obj(underscore)handle: the handle...

14/3,K/7 (Item 7 from file: 348)

DIALOG(R)File 348:EUROPEAN PATENTS

00572884

Method and apparatus for implementing a pseudo-LRU cache memory replacement scheme with a locking feature

Verfahren und Vorrichtung zur Durchfuhrung eines pseudo-LRU Cachespeicherersetzungsschemas mit einer Verriegelungsfunktion

Procede et dispositif de mise en oeuvre d'un schema de remplacement pseudo-LRU avec une fonction verrouillage

PATENT ASSIGNEE:
SUN MICROSYSTEMS, INC., (1392730), 2550 Garcia Avenue, Mountain View, CA 94043, (US), (applicant designated states: DE;FR;GB;IT;SE)
INVENTOR:
Malamy, Adam, 39 Wildwood Street, Winchester, Massachusetts 01890, (US)
Patel, Rajiv N., 3116 Whitesand Drive, San Jose, California 95148, (US)
Hayes, Norman M., 1121 Merrimac Drive, Sunnyvale, California 94087, (US)
LEGAL REPRESENTATIVE:
Wombwell, Francis (46021), Potts, Kerr & Co. 15, Hamilton Square, Birkenhead Merseyside L41 6BR, (GB)

PATENT (CC, No, Kind, Date): EP 568221 A1 931103 (Basic)
EP 568221 B1 981223
APPLICATION (CC, No, Date): EP 93302988 930419;
PRIORITY (CC, No, Date): US 875357 920429
DESIGNATED STATES: DE; FR; GB; IT; SE
INTERNATIONAL PATENT CLASS: **G06F-012/12**
ABSTRACT WORD COUNT: 139

LANGUAGE (Publication,Procedural,Application): English; English; English
FULLTEXT AVAILABILITY:

Available Text	Language	Update	Word Count
CLAIMS B	(English)	9852	1569
CLAIMS B	(German)	9852	1374
CLAIMS B	(French)	9852	1728
SPEC B	(English)	9852	4527
Total word count - document A			0
Total word count - document B			9198
Total word count - documents A + B			9198

INTERNATIONAL PATENT CLASS: **G06F-012/12**

...SPECIFICATION user locks cache memory line 1. Now all the cache memory lines that can be locked are locked. To prevent the cache memory from being **deadlocked** , the system clears the MRU bits. Only cache memory line 0 is available for replacement when all the other lines are locked. In step 10...

...0. The MRU bit for line 0 is not set by the cache controller since this would cause the composite mask to become "1111" causing **cache** memory **deadlock** .

In step 11, a **cache** hit occurs on line 1. The MRU bit for line 1 is set to "1" indicating that it has been used recently. Still, only cache ...

...the MRU bit for line 0 is set to "1". Unlike step 10, the setting of line 0's MRU bit will now not cause **deadlock** because additional lines have been unlocked.

As noted earlier, a distinctive advantage gained by utilizing the **locking** mechanism in the **cache** system of the present invention is the added intelligence provided to the **cache** replacement process. The **lock** bits are set by the application process thereby eliminating the

intelligence required to try to provide that knowledge at the cache controller level. One way to provide the request to **lock** certain **cache** memory lines is for the application program to program such request into the application program in the form of a predetermined command or subroutine call...

...compiling this program will recognize the command request and provide the proper code to execute the command.
System Programs, such as operating system routines, some **database** or window system routines may be used for controlling the locking as set forth in the present invention. Locking performed in the system programs boosts...

14/3,K/8 (Item 8 from file: 348)
DIALOG(R)File 348:EUROPEAN PATENTS

00566441
Method and apparatus for executing a distributed transaction in a distributed database
Verfahren und Vorrichtung um eine verteilte Transaktion in einer verteilten Datenbank auszufuhren
Procede et dispositif pour executer une transaction distribuee dans une base de donnees distribuee
PATENT ASSIGNEE:
ORACLE CORPORATION, (1640220), 500 Oracle Parkway, Redwood Shores, CA 94065, (US), (Proprietor designated states: all)
INVENTOR:
Hallmark, Gary, 1503 Alta Glen Drive, San Jose, CA 95125, (US)
BAMFORD, ROGER, 170 ROBERTA DRIVE, WOODSIDE, CA 94062, (US)
LEGAL REPRESENTATIVE:
Zenz, Joachim Klaus, Dipl.-Ing. et al (13445), Zenz, Helber, Hosbach & Partner, Patentanwalte, Huyssenallee 58-64, 45128 Essen, (DE)
PATENT (CC, No, Kind, Date): EP 567999 A2 931103 (Basic)
EP 567999 A3 940420
EP 567999 B1 990901
APPLICATION (CC, No, Date): EP 93106799 930427;
PRIORITY (CC, No, Date): US 876626 920430
DESIGNATED STATES: AT; BE; CH; DE; DK; ES; FR; GB; GR; IE; IT; LI; LU; MC; NL; PT; SE
INTERNATIONAL PATENT CLASS: **G06F-017/30**
ABSTRACT WORD COUNT: 217
NOTE:
Figure number on first page: 1

LANGUAGE (Publication,Procedural,Application): English; English; English
FULLTEXT AVAILABILITY:

Available Text	Language	Update	Word Count
CLAIMS A	(English)	EPABF1	610
SPEC A	(English)	EPABF1	6745
Total word count - document A			7355
Total word count - document B			0
Total word count - documents A + B			7355

INTERNATIONAL PATENT CLASS: **G06F-017/30**

...SPECIFICATION consistency.
Thompson, U. S. Patent 4,881,166, discloses a concurrence control that ensures the correct execution of multiple concurrent global transactions in a distributed **database** system along with independent concurrent execution of local transactions at each site. Thompson uses a two-phase commit protocol between the serves and the local **databases** to commit the updates performed by the global transactions on the global **database** . Thompson monitors possible inconsistency conditions between transactions, and prevents a transaction from executing as long as an inconsistency or **deadlock** is possible.
If the end time for transaction T does not fall between the start and

end time for any other involved transaction, then transaction T is allowed to execute, since no possibility of a **deadlock** or inconsistency may occur. Instead of logging the prepare and commit times for each transaction, only the start and end times of each transaction are logged. Although the method of Thompson does ensure read and write consistency in a distributed **database** , Thompson does not disclose any method of **locking** , and does not disclose row **level** **locking** capable of writing **locking** information as the transaction proceeds.

Ecklund, U. S. Patent 4,853,843, presents a system for merging virtual partitions on an objected-oriented, distributed **database** system following failure between sites accessing the **database** . Following restoration of site communication, the virtual partitions are merged to form a consistent merged **database** . Ecklund does not log prepare, commit and start times for each transaction, and does not disclose any method of locking data to ensure distributed read...

14/3,K/10 (Item 10 from file: 348)

00540969

Lock management in a multi-system shared data environment.

Verriegelungsverwaltung in einer Umgebung mit zwischen mehreren Systemen geteilten Daten.

Gestion de verrouillage dans un environnement de donnees partagees entre plusieurs systemes.

PATENT ASSIGNEE:
International Business Machines Corporation, (200120), Old Orchard Road, Armonk, N.Y. 10504, (US), (applicant designated states: DE;FR;GB)
INVENTOR:
Mohan, Chandrasekaran, 727 Portswood Drive, San Jose, California 95120, (US)
Narang, Inderpal S., 13778 Serra Oaks Court, Saratoga, California 95070, (US)
LEGAL REPRESENTATIVE:
Burt, Roger James, Dr. (52152), IBM United Kingdom Limited Intellectual Property Department Hursley Park, Winchester Hampshire SO21 2JN, (GB)

PATENT (CC, No, Kind, Date): EP 518639 A2 921216 (Basic)
EP 518639 A3 930818
APPLICATION (CC, No, Date): EP 92305320 920610;
PRIORITY (CC, No, Date): US 715505 910614
DESIGNATED STATES: DE; FR; GB
INTERNATIONAL PATENT CLASS: **G06F-009/46**
ABSTRACT WORD COUNT: 136

LANGUAGE (Publication,Procedural,Application): English; English; English
FULLTEXT AVAILABILITY:

Available Text	Language	Update	Word Count
CLAIMS A	(English)	EPABF1	678
SPEC A	(English)	EPABF1	7697
Total word count - document A			8375
Total word count - document B			0
Total word count - documents A + B			8375

INTERNATIONAL PATENT CLASS: **G06F-009/46**

...SPECIFICATION 30, 32 also includes a transaction manager (not shown) such as the IBM IMS/VS or the IBM CICS, that interfaces transactions with the respective **database** management systems. Each **DBMS** is capable of interfacing with N transactions. For simplicity, two such transactions 49 and 50 are illustrated interfacing with **DBMS** 36.

The **database** management systems can comprise available products such as the DB2 or IMS/VS systems available from the Assignee. It is assumed that the **DBMS** 's 36 and 37 include other functions, such as **database** recovery control. It is further assumed that abnormal and failure conditions in any of the components illustrated in Figure 2 are detected and either reportedthe lock request is called an L **lock** ; therefore, all

L- **type** **locks** denote transaction interest in a data resource. The lock request may be in a mode which is appropriate to the use to which the transaction...

...mode denotes that the transaction may update the resource by acquiring a lock in X-mode. U mode is incompatible with itself. U mode prevents **deadlocks** between transactions when they try to update the same resource after reading it.)

After acquiring the transaction **lock** , the **DBMS** checks its local **cache** to determine whether the resource resides therein. In the invention, it is assumed that the unit of ownership of a data resource by a transaction...

14/3,K/12 (Item 12 from file: 348)

DIALOG(R)File 348:EUROPEAN PATENTS

00460511

Dynamic load balancer for shared data base transaction processing.

Dynamischer Lastabgleicher fur gemeinschaftliche Datenbanktransaktionsverarbeitung.

Equilibreur dynamique de charge pour traitement des transactions dans une base de donnees partagee.

PATENT ASSIGNEE:
International Business Machines Corporation, (200120), Old Orchard Road, Armonk, N.Y. 10504, (US), (applicant designated states: DE;FR;GB)
INVENTOR:
Georgiadis, Leonidas, 38 Old Mill Road, Chappaqua, New York 10514, (US)
Nikolaou, Christos Nicolas, 5 Riverside Drive, Apt. 12E, New York, New York 10023, (US)
Wang, George Wei, 3140 Cedar Road, Yorktown Heights, New York 10598, (US)
LEGAL REPRESENTATIVE:
Teufel, Fritz, Dipl.-Phys. et al (11855), IBM Deutschland Informationssysteme GmbH, Patentwesen und Urheberrecht, 70548 Stuttgart , (DE)
PATENT (CC, No, Kind, Date): EP 459134 A2 911204 (Basic)
EP 459134 A3 930721
APPLICATION (CC, No, Date): EP 91106285 910419;
PRIORITY (CC, No, Date): US 516642 900430
DESIGNATED STATES: DE; FR; GB
INTERNATIONAL PATENT CLASS: **G06F-009/46**
ABSTRACT WORD COUNT: 104

LANGUAGE (Publication,Procedural,Application): English; English; English
FULLTEXT AVAILABILITY:

Available Text	Language	Update	Word Count
CLAIMS A	(English)	EPABF1	1123
SPEC A	(English)	EPABF1	6022
Total word count - document A			7145
Total word count - document B			0
Total word count - documents A + B			7145

INTERNATIONAL PATENT CLASS: **G06F-009/46**

...SPECIFICATION to the remaining routers.

In addition to the semi-dynamic feature of the present invention, the load balancer also reassigns transactions based on the transaction **type** . This is done to avoid **lock** contention and **deadlock** . For example, a given transaction **type** is likely to address the same memory and/or direct access storage device (DASD) addresses. If all of these types of transactions reside on the...

...iterative process based on a scheme of category weighted transaction types to incrementally improve load balancing in a system of computers which access a shared **data** **base** . More specifically, transaction types are divided in three subsets. The first of these is all transactions types such that if a single type is removed...

14/3,K/13 (Item 13 from file: 348)
DIALOG(R)File 348:EUROPEAN PATENTS
(c) 2004 European Patent Office. All rts. reserv.

00400232
Computerised database system.
Rechnergefuhrtes Datenbasissystem.
Base de donnees informatisee.
PATENT ASSIGNEE:
CROFTLANDS LIMITED, (1223360), Westaway Chambers, 39 Don Street, Jersey, St Helier, Channel Islands, (GB), (applicant designated states: AT;BE;CH;DE;DK;ES;FR;GB;GR;IT;LI;LU;NL;SE)
INVENTOR:
Dickerson, John, Flat 3, 4 Green Street, Mayfair, London, W1Y 3RG, (GB)
LEGAL REPRESENTATIVE:
Kennington, Eric Alasdair et al (50235), Beresford & Co. 2-5 Warwick Court High Holborn, London WC1R 5DJ, (GB)
PATENT (CC, No, Kind, Date): EP 394019 A2 901024 (Basic)
EP 394019 A3 930113
APPLICATION (CC, No, Date): EP 90304163 900418;
PRIORITY (CC, No, Date): GB 8908999 890420
DESIGNATED STATES: AT; BE; CH; DE; DK; ES; FR; GB; GR; IT; LI; LU; NL; SE
INTERNATIONAL PATENT CLASS: **G06F-015/40**
ABSTRACT WORD COUNT: 168
LANGUAGE (Publication,Procedural,Application): English; English; English
FULLTEXT AVAILABILITY:

Available Text	Language	Update	Word Count
CLAIMS A	(English)	EPABF1	2334
SPEC A	(English)	EPABF1	15362
Total word count - document A			17696
Total word count - document B			0
Total word count - documents A + B			17696

INTERNATIONAL PATENT CLASS: **G06F-015/40**

...SPECIFICATION order, regardless of their class.
This roll back provision is necessary for the following reason. Whenever an APR 53 applies an update statement to a **database** manager, e.g. the DB2 **database** manager 51, a lock is placed on the relevant resources which prevents all other APRs 53 from having access to those resources. Because a plurality of transactions are being applied in parallel, it is possible for a transaction not at the head of the sync queue for its **class** to obtain a **lock** on certain resources, and subsequently for the transaction at the head of the queue to require access to those same resources. This is a **deadlock** situation, since the transaction which has a lock on the resources will not release the lock until it passes through sync, and it cannot pass...

14/3,K/14 (Item 14 from file: 348)
DIALOG(R)File 348:EUROPEAN PATENTS
(c) 2004 European Patent Office. All rts. reserv.

00355760
Wait depth limited concurrency control method.
Gleichzeitigkeitssteuerungsverfahren mit Beschrankung der Warte-Tiefe.
Methode de commande de simultaneite avec limitation de profondeur d'attente.
PATENT ASSIGNEE:
International Business Machines Corporation, (200120), Old Orchard Road, Armonk, N.Y. 10504, (US), (applicant designated states: DE;FR;GB)
INVENTOR:
Franaszek, Peter A., T.J. Watson Research Center, P.O. Box 218,, Yorktown Heights, New York 10598, (US)
Robinson, John Timothy, 3314 North Deerfield Ave, Yorktown Heights New York 10598, (US)

Thomasian, Alexander, 17 Meadowbrook Road, Pleasantville New York 10570, (US)
LEGAL REPRESENTATIVE:
Jost, Ottokarl, Dipl.-Ing. (6092), IBM Deutschland Informationssysteme GmbH, Patentwesen und Urheberrecht, D-70548 Stuttgart, (DE)
PATENT (CC, No, Kind, Date): EP 377133 A2 900711 (Basic)
EP 377133 A3 921223
EP 377133 B1 950927
APPLICATION (CC, No, Date): EP 89122612 891207;
PRIORITY (CC, No, Date): US 294334 890105
DESIGNATED STATES: DE; FR; GB
INTERNATIONAL PATENT CLASS: **G06F-009/46**
ABSTRACT WORD COUNT: 168

LANGUAGE (Publication,Procedural,Application): English; English; English
FULLTEXT AVAILABILITY:

Available Text	Language	Update	Word Count
CLAIMS A	(English)	EPABF1	508
CLAIMS B	(English)	EPAB95	520
CLAIMS B	(German)	EPAB95	558
CLAIMS B	(French)	EPAB95	588
SPEC A	(English)	EPABF1	1931
SPEC B	(English)	EPAB95	2088
Total word count - document A			2439
Total word count - document B			3754
Total word count - documents A + B			6193

INTERNATIONAL PATENT CLASS: **G06F-009/46**

...SPECIFICATION of concurrency control is needed in order to avoid problems when two or more users attempt to update a field of a record in the **database** on the basis of an initial value of that field. One approach to concurrency control is known as locking. Another is known as time-stamping...

...record by issuing a request to a system component called the lock manager. If a transaction holds an exclusive lock on some object, say, a **database** record, then no other transaction can acquire a **lock** of any **type** on that object until the first transaction releases its lock. Any transaction that intends to update a record must first acquire a lock on it...

...available and the lock can be granted. While this locking protocol solves the lost update problem, it introduces two others. One is the problem of **deadlock** , in which two or more transactions are in a simultaneous wait state, each waiting for the others to release a lock required for it to...

...are being processed concurrently) is that many or even most of these transactions can be waiting at a given time, even without the presence of **deadlock** . Increasing the **level** of concurrency (the number of transactions attempting to proceed simultaneously) can actually reduce the number doing useful work (i.e., not waiting or in **deadlock**) at a given time.

The problem of **deadlock** has been extensively studied. In general, the lock manager must be capable of detecting the occurrence of **deadlocks** and resolve them. Resolving the **deadlock** amounts to choosing one of the locked transactions and rolling it back. This process involves terminating the transaction and undoing all its updates and releasing its locks so that the resources concerned can be allocated to other transactions.

The general problems associated with concurrency in **database** transactions is considered in more detail by C. J. Date at Chapter 3, "Concurrency", An Introduction to **Database** Systems, Vol. II, Addison-Wesley Publishing Company (1983). The reader is referred to that text for more information on the various concurrency problems and protocols used, especially in the **locking** **type** concurrency controls.

A running priority (RP) concurrency control (CC) is described in an article entitled "Limitations of Concurrency in Transaction Processing" by P. A. Franaszek and J. T. Robinson, published in ACM Transactions on **Database** Systems 10, March 1985, pp. 1 to 28. This method results in improved performance compared to standard locking because it approximates "essential blocking" by having...

...SPECIFICATION of concurrency control is needed in order to avoid problems when two or more users attempt to update a field of a record in the **database** on the basis of an initial value of that field. One approach to concurrency control is known as locking. Another is known as timestamping or...

...record by issuing a request to a system component called the lock manager. If a transaction holds an exclusive lock on some object, say, a **database** record, then no other transaction can acquire a **lock** of any **type** on that object until the first transaction releases its lock. Any transaction that intends to update a record must first acquire a lock on it...

...available and the lock can be granted. While this locking protocol solves the lost update problem, it introduces two others. One is the problem of **deadlock** , in which two or more transactions are in a simultaneous wait state, each waiting for the others to release a lock required for it to...

...are being processed concurrently) is that many or even most of these transactions can be waiting at a given time, even without the presence of **deadlock** . Increasing the **level** of concurrency (the number of transactions attempting to proceed simultaneously) can actually reduce the number doing useful work (i.e., not waiting or in **deadlock**) at a given time.

The problem of **deadlock** has been extensively studied. In general, the lock manager must be capable of detecting the occurrence of **deadlocks** and resolve them. Resolving the **deadlock** amounts to choosing one of the locked transactions and rolling it back. This process involves terminating the transaction and undoing all its updates and releasing its locks so that the resources concerned can be allocated to other transactions.

The general problems associated with concurrency in **database** transactions is considered in more detail by C. J. Date at Chapter 3, "Concurrency", An Introduction to **Database** Systems, Vol. II, Addison-Wesley Publishing Company (1983). The reader is referred to that text for more information on the various concurrency problems and protocols used, especially in the **locking** **type** concurrency controls.

A running priority (RP) concurrency control (CC) is described in an article entitled "Limitations of Concurrency in Transaction Processing" by P. A. Franaszek and J. T. Robinson, published in ACM Transactions on **Database** Systems 10, March 1985, pp. 1 to 28. This method results in improved performance compared to standard locking because it approximates "essential blocking" by having...

14/3,K/15 (Item 15 from file: 348)
DIALOG(R)File 348:EUROPEAN PATENTS

00284777
Distributed file management system.
Verwaltungssystem fur verteilte Datei.
Systeme de gestion pour fichier distribue.
PATENT ASSIGNEE:
International Business Machines Corporation, (200120), Old Orchard Road, Armonk, N.Y. 10504, (US), (applicant designated states: DE;FR;GB)
INVENTOR:
Johnson, Donovan William, 212 Esparada, Georgetown Texas 78628, (US)
Shaheen-Gouda, Amal Ahmed, 11502 Sweetshade Lane, Austin Texas 78759, (US)

Smith, Todd Allen, 1802 Apricot Glen, Austin Texas 78746, (US)
LEGAL REPRESENTATIVE:
Burt, Roger James, Dr. et al (52152), IBM United Kingdom Limited
Intellectual Property Department Hursley Park, Winchester Hampshire
SO21 2JN, (GB)
PATENT (CC, No, Kind, Date): EP 278313 A2 880817 (Basic)
EP 278313 A3 901031
EP 278313 B1 940810
APPLICATION (CC, No, Date): EP 88101087 880126;
PRIORITY (CC, No, Date): US 14900 870213
DESIGNATED STATES: DE; FR; GB
INTERNATIONAL PATENT CLASS: **G06F-009/46** ; **G06F-015/40**
ABSTRACT WORD COUNT: 133

LANGUAGE (Publication,Procedural,Application): English; English; English
FULLTEXT AVAILABILITY:

Available Text	Language	Update	Word Count
CLAIMS A	(English)	EPBBF1	886
CLAIMS B	(English)	EPBBF1	1117
CLAIMS B	(German)	EPBBF1	1001
CLAIMS B	(French)	EPBBF1	1357
SPEC A	(English)	EPBBF1	14054
SPEC B	(English)	EPBBF1	14046
Total word count - document A			14940
Total word count - document B			17521
Total word count - documents A + B			32461

INTERNATIONAL PATENT CLASS: **G06F-009/46** ...

... **G06F-015/40**

...CLAIMS system, by means of the remote procedure call request, for serialising access to a list of client data processing systems having current access to the **file** ;
locking the first lock by means of an operation executing at the server data processing system and requiring access to the **file** at the server data processing system; and
unlocking the first lock while maintaining the third lock before sending the remote procedure call to the client...requiring a lock on the first lock after the remote procedure call is sent.
17. A method, in a data processing system, of preventing a **deadlock** between a first lock that serializes access to data in a **file** at a server data processing system and a second lock that serializes access to data, corresponding to the **file** , in a cache in a client data processing system, the method comprising the steps of:
locking the first lock by means of an operation executing at the server data processing system and requiring access to the **file** at the server data processing system;
locking the second lock by means of an operation executing at the client data processing system requiring access to the data in the **cache** ;
unlocking the second **lock** by means of an operation at the client data processing system if the operation generates a remote procedure call from the client data processing system...

14/3,K/16 (Item 16 from file: 348)
DIALOG(R)File 348:EUROPEAN PATENTS

00284775
Distributed file and record locking.
Sperrung von verteilten Dateien und Aufzeichnungen.
Bloquage de fichiers distribues et d'enregistrements.
PATENT ASSIGNEE:
International Business Machines Corporation, (200120), Old Orchard Road, Armonk, N.Y. 10504, (US), (applicant designated states: DE;FR;GB)
INVENTOR:

Henson, Larry William, 4207 Avenue G, Austin Texas 78751, (US)
Shaheen-Gouda, Amal Ahmed, 11502 Sweetshade Lane, Austin Texas 78759, (US)
Smith, Todd Allen, 1802 Apricot Glen, Austin Texas 78746, (US)
LEGAL REPRESENTATIVE:
Burt, Roger James, Dr. et al (52152), IBM United Kingdom Limited Intellectual Property Department Hursley Park, Winchester Hampshire SO21 2JN, (GB)
PATENT (CC, No, Kind, Date): EP 278312 A2 880817 (Basic)
EP 278312 A3 901024
EP 278312 B1 941026
APPLICATION (CC, No, Date): EP 88101085 880126;
PRIORITY (CC, No, Date): US 14891 870213
DESIGNATED STATES: DE; FR; GB
INTERNATIONAL PATENT CLASS: **G06F-009/46 ; G06F-015/40**
ABSTRACT WORD COUNT: 239

LANGUAGE (Publication,Procedural,Application): English; English; English
FULLTEXT AVAILABILITY:

Available Text	Language	Update	Word Count
CLAIMS A	(English)	EPBBF1	825
CLAIMS B	(English)	EPBBF1	851
CLAIMS B	(German)	EPBBF1	802
CLAIMS B	(French)	EPBBF1	937
SPEC A	(English)	EPBBF1	14831
SPEC B	(English)	EPBBF1	14797
Total word count - document A			15656
Total word count - document B			17387
Total word count - documents A + B			33043

INTERNATIONAL PATENT CLASS: **G06F-009/46** ...

... **G06F-015/40**

...SPECIFICATION of this architecture. First, processes may have to use remote procedure calls (RPC)s to set or test locks. These RPCS will run on the **file** 's server. Second, when the synch mode of a **file** changes, the **file** 's lock list may have to be moved from the client to the server or vice-versa. The entries of an inode's lock table correspond to locks over segments of the inode's **file** . To represent a lock, a lock list entry must contain information identifying the range of bytes **locked** , the **type** of **lock** (read or write), the owner of the lock.
In standalone UNIX operating system a process that tries to establish a lock may have to wait for an existing lock to clear first. Before waiting (going to sleep) the process must check the sleep list to insure that no **deadlock** will occur if it does wait. A waiting process has its proc table use the W...

...SPECIFICATION of this architecture. First, processes may have to use remote procedure calls (RPC)s to set or test locks. These RPCs will run on the **file** ' s server. Second, when the synch mode of a **file** changes, the **file** 's lock list may have to be moved from the client to the server or vice-versa. The entries of an inode's lock table correspond to locks over segments of the inode's **file** . To represent a lock, a lock list entry must contain information identifying the range of bytes **locked** , the **type** of **lock** (read or write), the owner of the lock.
In standalone UNIX operating system a process that tries to establish a lock may have to wait for an existing lock to clear first. Before waiting (going to sleep) the process must check the sleep list to insure that no **deadlock** will occur if it does wait. A waiting process has its proc table use the W...

14/3,K/18 (Item 1 from file: 349)
DIALOG(R)File 349:PCT FULLTEXT

01139749 **Image available**

METADATA BASED FILE SWITCH AND SWITCHED FILE SYSTEM
COMMUTATEUR DE FICHIER UTILISANT DES METADONNEES ET SYSTEME FICHIER COMMUTE

Patent Applicant/Assignee:
Z-FORCE INC, 24461 Ridge Route Drive, Suite 100, Laguna Hills, CA 92653, US, US (Residence), US (Nationality)
Inventor(s):
MILOUSHEV Vladimir, 35 Terraza Del Mar, Dana Point, CA 92629, US,
NICKOLOV Peter, 70 Calais Street, Laguna Niguel, CA 92677, US,
Legal Representative:
WILLIAMS Gary S (et al) (agent), Morgan, Lewis & Bockius LLP, 3300 Hillview Avenue, Palo Alto, CA 94304, US,
Patent and Priority Information (Country, Number, Date):
Patent: WO 200461605 A2 20040722 (WO 0461605)
Application: WO 2003US41202 20031218 (PCT/WO US03041202)
Priority Application: US 2003336704 20030102; US 2003336784 20030102; US 2003336832 20030102; US 2003336833 20030102; US 2003336834 20030102; US 2003336835 20030102
Designated States:
(Protection type is "patent" unless otherwise stated - for applications prior to 2004)
AE AG AL AM AT AU AZ BA BB BG BR BY BZ CA CH CN CO CR CU CZ DE DK DM DZ EC EE EG ES FI GB GD GE GH GM HR HU ID IL IN IS JP KE KG KP KR KZ LC LK LR LS LT LU LV MA MD MG MK MN MW MX MZ NI NO NZ OM PG PH PL PT RO RU SC SD SE SG SK SL SY TJ TM TN TR TT TZ UA UG UZ VC VN YU ZA ZM ZW
(EP) AT BE BG CH CY CZ DE DK EE ES FI FR GB GR HU IE IT LU MC NL PT RO SE SI SK TR
(OA) BF BJ CF CG CI CM GA GN GQ GW ML MR NE SN TD TG
(AP) BW GH GM KE LS MW MZ SD SL SZ TZ UG ZM ZW
(EA) AM AZ BY KG KZ MD RU TJ TM
Publication Language: English
Filing Language: English
Fulltext Word Count: 31089

Main International Patent Class: **G06F**
Fulltext Availability:
Detailed Description

Detailed Description

... switch 200 and refuse the requests 2102, 2104 and 2105 from the file switch 2106. The file server 202 satisfies the request 2103 from the **file** switch 2106 and refuses the request 2113 from the file switch 200. As a result, from the standpoint ofa file switch, both aggregated transactions will...

...semantics state that one client should succeed and all others should fail.

[0323] One skilled in the art will recognize that this situation is a **classic deadlock** problem. Although the resource that both clients requested (i.e., the aggregated **file**) is available and can be granted to one of the clients easily,@ none of the clients is able to acquire it (i.e., write to the **file**).

Implicit Locking
[03241, Network **file** protocols typically provide **file - level locking** and byte-range **locking** in order to synchronize multiple clients that try to write to the same **file** and the same area within a **file** . When locking is used consistently by all clients, there is no need for additional synchronization in order to avoid inconsistent data being written to different mirrors -of the same **file** ; however, not all **file** client applications use the locking mechanism consistently.

[03251 Implicit locking allows a client to write data into a locked byte range while sharing the same **file** with other clients. While a client holds a lock on a byte range in a, **file** , it is the only client that is allowed to write data into that portion of the file. Other clients can not read or write data...

14/3,K/19 (Item 2 from file: 349)
DIALOG(R)File 349:PCT FULLTEXT

00973263 **Image available**
A METHOD FOR CONCURRENCY CONTROL FOR A SECUNDARY INDEX
PROCEDE DE CONTROLE DE CONCURRENCE POUR INDEX SECONDAIRE
Patent Applicant/Assignee:
TELEFONAKTIEBOLAGET LM ERICSSON (publ), S-126 25 Stockholm, SE, SE (Residence), SE (Nationality), (For all designated states except: US)
Patent Applicant/Inventor:
RONSTROM Mikael, Hagerstensvagen 119, S-126 48 Stockholm, SE, SE (Residence), SE (Nationality), (Designated only for: US)
Legal Representative:
DR LUDWIG BRANN PATENTBYRA AB (agent), Box 171 92, S-104 62 Stockholm, SE ,
Patent and Priority Information (Country, Number, Date):
Patent: WO 200303251 A1 20030109 (WO 0303251)
Application: WO 2001SE1479 20010628 (PCT/WO SE0101479)
Priority Application: WO 2001SE1479 20010628
Designated States:
(Protection type is "patent" unless otherwise stated - for applications prior to 2004)
AE AG AL AM AT AU AZ BA BB BG BR BY BZ CA CH CN CO CR CU CZ DE DK DM DZ EE ES FI GB GD GE GH GM HR HU ID IL IN IS JP KE KG KP KR KZ LC LK LR LS LT LU LV MA MD MG MK MN MW MX MZ NO NZ PL PT RO RU SD SE SG SI SK SL TJ TM TR TT TZ UA UG US UZ VN YU ZA ZW
(EP) AT BE CH CY DE DK ES FI FR GB GR IE IT LU MC NL PT SE TR
(OA) BF BJ CF CG CI CM GA GN GW ML MR NE SN TD TG
(AP) GH GM KE LS MW MZ SD SL SZ TZ UG ZW
(EA) AM AZ BY KG KZ MD RU TJ TM
Publication Language: English
Filing Language: English
Fulltext Word Count: 5875

Main International Patent Class: **G06F-017/30**
Fulltext Availability:
Detailed Description

Detailed Description
... in order to ensure that no conflicting operations are performed simultaneously on the same record or index record, which otherwise could cause errors in the **database** and the index. The level of concurrency depends on the **locking** strategy that is used. In a pessimistic locking strategy concurrency is only allowed between two read operations, while there is no concurrency between two write...

...between a read operation and a write operation. In an environment where secondary indexes are used, which are stored in separate index tables or other **types** of storage containers, the pessimistic **locking** strategy gives bad concurrency and, at the same time, causes frequent **deadlocks** . A **deadlock** can occur when two different operations have acquired locks that stop the other operation from completing. The **deadlock** can be resolved by aborting one of the operations. Many of the operations described above give rise to a series of operations that require locks...

...the table and others from the secondary index it is easily understood that a traditional locking strategy, such as the pessimistic locking strategy, often causes **deadlocks** . **Deadlocks** are naturally highly undesirable and there is thus a demand for a **locking** strategy that to a high **degree** prevents **deadlocks** f rom occurring.

U.S. Patent 5,.280,612 describes a database system that aims to increase the concurrency in the system and thereby reduce...above.

An object of the present invention is thus to provide a method and arrangements for concurrency control for secondary indexes that to a high **degree** prevents **deadlocks** from occurring due to operations performed on the secondary index.

The object of the present invention is achieved by means of a method as stated of a method for concurrency control for a secondary index wherein many **deadlocks** are avoided by means of a selective locking strategy and by means of allowing index records of the secondary index to contain some keys to...

14/3,K/20 (Item 3 from file: 349)
DIALOG(R)File 349:PCT FULLTEXT

00943734 **Image available**
APPARATUS FOR MONITORING PERFORMANCE OF DATABASE AND METHOD THEREOF
PROCEDE ET DISPOSITIF PERMETTANT DE CONTROLER LE RENDEMENT D'UNE BASE DE DONNEES
Patent Applicant/Assignee:
EXEM LTD, 4Th Floor, Future-Flow Bldg., 1538-9 Seocho-dong, Seocho-gu, Seoul 137-070, KR, KR (Residence), KR (Nationality), (For all designated states except: US)
Patent Applicant/Inventor:
CHO Chong-Arm, 105-1301 Seongdong-Maul-Gangnam-Village, 84 Seongbook-ri, Sooji-eup, Yongin-shi 449-840, Gyeonggi-do, KR, KR (Residence), KR (Nationality), (Designated only for: US)
Legal Representative:
PARK Young-Il (agent), HyundaiI Life Insurance Bldg., 5F., 649-14 Yoksam-dong, Gangnam-gu, Seoul 135-080, KR,
Patent and Priority Information (Country, Number, Date):
Patent: WO 200277879 A1 20021003 (WO 0277879)
Application: WO 2002KR491 20020322 (PCT/WO KR0200491)
Priority Application: KR 200115453 20010324
Designated States:
(Protection type is "patent" unless otherwise stated - for applications prior to 2004)
AE AG AL AM AT AU AZ BA BB BG BR BY BZ CA CH CN CO CR CU CZ DE DK DM DZ
EC EE ES FI GB GD GE GH GM HR HU ID IL IN IS JP KE KG KP KZ LC LK LR LS
LT LU LV MA MD MG MK MN MW MX MZ NO NZ OM PH PL PT RO RU SD SE SG SI SK
SL TJ TM TN TR TT TZ UA UG US UZ VN YU ZA ZM ZW
(EP) AT BE CH CY DE DK ES FI FR GB GR IE IT LU MC NL PT SE TR
(OA) BF BJ CF CG CI CM GA GN GQ GW ML MR NE SN TD TG
(AP) GH GM KE LS MW MZ SD SL SZ TZ UG ZM ZW
(EA) AM AZ BY KG KZ MD RU TJ TM
Publication Language: English
Filing Language: Korean
Fulltext Word Count: 8595

Main International Patent Class: **G06F-017/40**
Fulltext Availability:
Detailed Description

Detailed Description
... database level, a programming (i.e., a database session) level, and an SQL level is stored in a data dictionary 18 of the SGA 16.

Database performance data of a whole **database** level is a start point for **database** monitoring and may be classified as follows.

as for input/output (1/0)-related information, performance statistics data, i.e., logical read, physical read, and

direct read, etc., and
waiting event data, i.e., DB **file** sequential read and DB **file** scattered read, etc.;
as for SQL execution performance information at a whcle **database** level,
performance statistics data, i.e., user calls, recursive calls, parse count, and execution count, etc., and
waiting even data, i.e., latch free, library cache pin, and library 9
cache **lock** , etc.;
as for **lock** -related information,
performance statistics data, i.e., enqueue waits and equeue **deadlocks** , and
waiting event data, i.e., enqueue;
as for sort-related information,
performance statistics data, i.e., sort (e.g., memory, disk, and rows), and
waiting event data, i.e., DB **file** scattered read and direct path io read;and
as for response-time-related performance information,
performance statistics data, i.e., recursive CPU usage, CPU used by this session, parse time CPU, and parse time elapsed, and
waiting event data, i.e., all waiting information.

As described above, the **database** performance data of a whole **database** level is classified into performance statistics data and waiting
event data. The performance statistics data is an index which the Oracle database provides to track...

14/3,K/22 (Item 5 from file: 349)
DIALOG(R)File 349:PCT FULLTEXT

00912779 **Image available**
METHOD AND SYSTEM FOR RESPONDING TO FILE SYSTEM REQUESTS
PROCEDE ET SYSTEME DE REPONSE A DES DEMANDES DE SYSTEME DE FICHIER
Patent Applicant/Assignee:
SPINNAKER NETWORKS INC, 301 Alpha Drive, Pittsburgh, PA 15238, US, US (Residence), US (Nationality), (For all designated states except: US)
Patent Applicant/Inventor:
KAZAR Michael L, 1233 Shady Avenue, Pittsburgh, PA 15232, US, US (Residence), US (Nationality), (Designated only for: US)
NYDICK Daniel S, 2569 Barnwood Drive, Wexford, PA 15090, US, US (Residence), US (Nationality), (Designated only for: US)
SANZI Richard N Jr, 2659 Ravenwood Court, Wexford, PA 15090, US, US (Residence), US (Nationality), (Designated only for: US)
GOHH Fred, 2353 West Gate Drive, Pittsburgh, PA 15237, US, US (Residence), CA (Nationality), (Designated only for: US)
BIANCHINI Ronald P Jr, 326 Olde Chapel Trail, Pittsburgh, PA 15238, US, US (Residence), US (Nationality), (Designated only for: US)
SPEISER Benjamin, 138 Westland Drive, Pittsburgh, PA 15217, US, US (Residence), US (Nationality), (Designated only for: US)
Legal Representative:
SCHWARTZ Ansel M (agent), Suite 304, 201 N. Craig Street, Pittsburgh, PA 15213, US,
Patent and Priority Information (Country, Number, Date):
Patent: WO 200246931 A1 20020613 (WO 0246931)
Application: WO 2001US46792 20011205 (PCT/WO US0146792)
Priority Application: US 2000732121 20001207
Designated States:
(Protection type is "patent" unless otherwise stated - for applications prior to 2004)
AE AG AL AM AT AU AZ BA BB BG BR BY BZ CA CH CN CO CR CU CZ DE DK DM DZ
EE ES FI GB GD GE GH GM HR HU ID IL IN IS JP KE KG KP KR KZ LC LK LR LS
LT LU LV MA MD MG MK MN MW MX MZ NO NZ PL PT RO RU SD SE SG SI SK SL TJ
TM TR TT TZ UA UG US UZ VN YU ZA ZW

(EP) AT BE CH CY DE DK ES FI FR GB GR IE IT LU MC NL PT SE TR
(OA) BF BJ CF CG CI CM GA GN GQ GW ML MR NE SN TD TG
(AP) GH GM KE LS MW MZ SD SL SZ TZ UG ZM ZW
(EA) AM AZ BY KG KZ MD RU TJ TM
Publication Language: English
Filing Language: English
Fulltext Word Count: 10846

Main International Patent Class: **G06F-012/00**
International Patent Class: **G06F-012/08**
Fulltext Availability:
Detailed Description

Detailed Description
... with the specified deny modes (no other readers, no other writers, neither or both). The output parameters include the number of names consumed, the optional **file** attributes, and the oplock returned, if any (the desired oplock is specified along with the other locking. mode input ,parameters).

-2 0
The remote procedure...

...system 10, retransmissions in the case of message loss, flow control in the case of network congestion, and resource isolation on the server to prevent **deadlocks** when one **class** of request tries to consume resources required by the server to process the earlier received requests. Resource priorities are associated with calls to ensure that...

14/3,K/23 (Item 6 from file: 349)
DIALOG(R)File 349:PCT FULLTEXT

00868212
DATA MANAGEMENT APPLICATION PROGRAMMING INTERFACE FOR A PARALLEL FILE SYSTEM
INTERFACE DE PROGRAMMATION D'UNE APPLICATION DE GESTION DES DONNEES POUR UN SYSTEME DE FICHIER PARALLELES
Patent Applicant/Assignee:
INTERNATIONAL BUSINESS MACHINES CORPORATION, Orchard Road, Armonk, NY, US , US (Residence), US (Nationality), (For all designated states except: US)
Patent Applicant/Inventor:
LOY Irit, 6 Shunamit Street, 34562 Haifa, IL, IL (Residence), IL (Nationality), (Designated only for: US)
MARBERG John, 31 Vitkin Street, 34754 Haifa, IL, IL (Residence), IL (Nationality), (Designated only for: US)
SHMUELI Boaz, 6340 Aurelia Street, Apt. 1, Pittsburgh, PA 15206, US, US (Residence), IL (Nationality), (Designated only for: US)
YEHUDAI Zvi, 6A Hanasi Avenue, 34645 Haifa, IL, IL (Residence), IL (Nationality), (Designated only for: US)
CURRAN Robert, 8 Hedgerow Court, West Hurley, NY 12491, US, US (Residence), US (Nationality), (Designated only for: US)
HASKIN Roger, 13585 Struzenberg Court, Morgan Hill, CA 95037, US, US (Residence), US (Nationality), (Designated only for: US)
SCHMUCK Frank, 406 Union Avenue, Campbell, CA 95008, US, US (Residence), DE (Nationality), (Designated only for: US)
WYLLIE James, 18392 Chadbourne Lane, Monte Sereno, CA 95030, US, US (Residence), US (Nationality), (Designated only for: US)
Legal Representative:
COLB Sanford T (et al) (agent), Sanford T. Colb & Co., P.O. Box 2273, 76122 Rehovot, IL,
Patent and Priority Information (Country, Number, Date):
Patent: WO 200201410 A1 20020103 (WO 0201410)
Application: WO 2001IL560 20010619 (PCT/WO IL0100560)

Priority Application: US 2000214127 20000626
Designated States:
(Protection type is "patent" unless otherwise stated - for applications prior to 2004)
AE AG AL AM AT AU AZ BA BB BG BR BY BZ CA CH CN CO CR CU CZ DE DK DM DZ EC EE ES FI GB GD GE GH GM HR HU ID IL IN IS JP KE KG KP KR KZ LC LK LR LS LT LU LV MA MD MG MK MN MW MX MZ NO NZ PL PT RO RU SD SE SG SI SK SL TJ TM TR TT TZ UA UG US UZ VN YU ZA ZW
(EP) AT BE CH CY DE DK ES FI FR GB GR IE IT LU MC NL PT SE TR
(OA) BF BJ CF CG CI CM GA GN GW ML MR NE SN TD TG
(AP) GH GM KE LS MW MZ SD SL SZ TZ UG ZW
(EA) AM AZ BY KG KZ MD RU TJ TM
Publication Language: English
Filing Language: English
Fulltext Word Count: 13368

Main International Patent Class: **G06F-017/30**
Fulltext Availability:
Detailed Description

Detailed Description
... the access is held by the DM 1 5 application. Conversely, the DM application must be prevented from acquiring an access right while a conflicting **file** operation is in progress. Preferably, these access rights are implemented using the intemal locking mechanisms of PFS 28, such as the GPFS locking mechanisms described in the above-mentioned patents by Sclimuck et al.

DM access rights in cluster 20 are preferably treated as an additional **file** lock in the hierarchy of locks acquired during **file** access. This approach enables acquiring and releasing access rights efficiently, using existing, highly-optimized locking mechanisms of the PFS.
This additional lock is referred to herein as the "DM lock." The **lock** characteristics are affected by the **type** of access (shared or exclusive) and the **type** of thread acquiring the **lock** (**file** operation thread or data management operation tliread). Existing **file** locks (such as those described by Schmuck et al.) cannot be used for this purpose, since DM access rights are held across multiple kernel calls and can be shared aniong DM application tlireads without going through the kemel. The existing **file** locks are still required to synchronize access to **file** data, even while a DM access right is held. Preferably, to prevent **deadlocks** , the DM lock is acquired before any other locks in the **file** locking hierarchy.

Table 1 below is a lock conflict table that defines DM access rights semantics in cluster 20. Four lock modes are used.

1...

14/3,K/24 (Item 7 from file: 349)
DIALOG(R)File 349:PCT FULLTEXT

00826039 **Image available**
ACTIVE COOPERATION DEADLOCK DETECTION SYSTEM/METHOD IN A DISTRIBUTED DATABASE NETWORK
SYSTEME ET UN PROCEDE DE BLOCAGE/DEBLOCAGE PAR COOPERATION ACTIVE DANS UN RESEAU A BASE DE DONNEES COMMUNE
Patent Applicant/Assignee:
TELEFONAKTIEBOLAGET LM ERICSSON (publ), S-126 25 Stockholm, SE, SE (Residence), SE (Nationality)
Inventor(s):
WIGER Ulf T, Kungsklippevagen 9, S-141 40 Huddinge, SE,
ARTS Thomas H J J, Svensborgsvagen 12, S-141 38 Huddinge, SE,
Legal Representative:
JOHANSSON H (et al) (agent), Albihns Stockholm AB, P.O. Box 5581, S-114 85 Stockholm, SE,

Patent and Priority Information (Country, Number, Date):
  Patent: WO 200159568 A2-A3 20010816 (WO 0159568)
  Application: WO 2001SE265 20010209 (PCT/WO SE0100265)
  Priority Application: US 2000502672 20000211
Designated States:
(Protection type is "patent" unless otherwise stated - for applications prior to 2004)
  AE AG AL AM AT AU AZ BA BB BG BR BY BZ CA CH CN CR CU CZ DE DK DM DZ EE ES FI GB GD GE GH GM HR HU ID IL IN IS JP KE KG KP KR KZ LC LK LR LS LT LU LV MA MD MG MK MN MW MX MZ NO NZ PL PT RO RU SD SE SG SI SK SL TJ TM TR TT TZ UA UG UZ VN YU ZA ZW
  (EP) AT BE CH CY DE DK ES FI FR GB GR IE IT LU MC NL PT SE TR
  (OA) BF BJ CF CG CI CM GA GN GW ML MR NE SN TD TG
  (AP) GH GM KE LS MW MZ SD SL SZ TZ UG ZW
  (EA) AM AZ BY KG KZ MD RU TJ TM
Publication Language: English
Filing Language: English
Fulltext Word Count: 14371

Main International Patent Class: **G06F-009/46**
Fulltext Availability:
  Claims

Claim
... second data objects; and
  after completing its task, the first client surrendering its locks on the first
  and second data ob'
  Jects.
  16 A distributed **database** network comprising:
  first and second clients for perfoilming first and second transactions, respectively;
  a plurality of data objects which may be accessed by said first...

...client including data indicative of an event or data object that the second client is waiting for.

  17 The network of claim 16, wherein said **deadlock** detection system causes one of the first and second clients to surrender a lock on a data object in order to initiate resolution of the **deadlock** .

  18 A **deadlock** detection system comprising:
  first and second processes for performing first and second transactions, respectively, each transaction utilizing at least one common data object stored in a
  replicated **database** system; and
  at least said first process comprising a **deadlock** detection system including dependency **information** **stored** relating to the first and second transactions, for detecting a **deadlock** in which said first and second clients are involved.
  19 The system of claim 18, wherein each of said first and second processes comprises a **deadlock** detection system, each including a graph or table having information stored therein relating to the first and second transactions, for detecting a **deadlock** in which said first and second clients are involved, and wherein the graph or table of each process is updated based upon information received from...

...of claim 18, wherein said dependency information includes only information relating to said first transaction.

  22 A method of a first client in a distributed **database** network determining whether to send infori-nation relating to a lock on a data object to a second client, the
  method comprising the steps of...

14/3,K/39 (Item 22 from file: 349)
DIALOG(R)File 349:PCT FULLTEXT

00764271 **Image available**

METHOD AND ARTICLE OF MANUFACTURE FOR ISOLATING DATA WITHIN A COMPUTER PROGRAM

PROCEDE ET ARTICLE MANUFACTURE POUR ISOLATION DE DONNEES DANS UN PROGRAMME INFORMATIQUE

Patent Applicant/Assignee:
ACCENTURE LLP, 1661 Page Mill Road, Palo Alto, CA 94304, US, US (Residence), US (Nationality), (For all designated states except: US)

Patent Applicant/Inventor:
BALON Richard E, 197 Hawkins Circle, Wheaton, IL 60187, US, US (Residence), US (Nationality), (Designated only for: US)
MALIK Asif, 1216 N. Astor #3A, Chicago, IL 60610, US, US (Residence), US (Nationality), (Designated only for: US)
WARGIN Jeffrey M, 161 North Clark Street, Chicago, IL 60601, US, US (Residence), US (Nationality), (Designated only for: US)
JACKOWSKI Michael A, 3602 Franklin Ct., Crystal Lake, Il 60014, US, US (Residence), US (Nationality), (Designated only for: US)
KENNEDY Richard C, 161 North Clark Street, Chicago, IL 60601, US, US (Residence), US (Nationality), (Designated only for: US)
NAVICKAS Eduardo, 525 West Hawthorne Place, Chicago, IL 60657, US, US (Residence), -- (Nationality), (Designated only for: US)

Legal Representative:
STEPHENS L Keith (agent), Hickman Stephens Coleman & Hughes, LLP, P.O. Box 52037, Palo Alto, CA 94303-0746, US,

Patent and Priority Information (Country, Number, Date):
Patent: WO 200077698 A2-A3 20001221 (WO 0077698)
Application: WO 2000US12351 20000504 (PCT/WO US0012351)
Priority Application: US 99305816 19990504

Designated States:
(Protection type is "patent" unless otherwise stated - for applications prior to 2004)
AE AG AL AM AT AU AZ BA BB BG BR BY CA CH CN CR CU CZ DE DK DM DZ EE ES FI GB GD GE GH GM HR HU ID IL IN IS JP KE KG KP KR KZ LC LK LR LS LT LU LV MA MD MG MK MN MW MX NO NZ PL PT RO RU SD SE SG SI SK SL TJ TM TR TT TZ UA UG US UZ VN YU ZW
(EP) AT BE CH CY DE DK ES FI FR GB GR IE IT LU MC NL PT SE
(OA) BF BJ CF CG CI CM GA GN GW ML MR NE SN TD TG
(AP) GH GM KE LS MW SD SL SZ TZ UG ZW
(EA) AM AZ BY KG KZ MD RU TJ TM

Publication Language: English
Filing Language: English
Fulltext Word Count: 28267

Main International Patent Class: **G06F-009/44**
International Patent Class: **G06F-009/46**

Fulltext Availability:
Detailed Description

Detailed Description

... to both read and update records concurrently. This situation is treated as exceptional processing rather than normal processing. Locks are not actually placed on the **database** at read time. A timestamp mechanism is used at time of update or delete to ensure that another user has not modified or deleted the record since you last read the record.

A preferred embodiment of the present invention uses an optimistic locking approach to concurrency control. This ensures **database** integrity as well as the low overhead associated with this form of locking. Other benefits to this method are increased availability of records to multiple users, and a minimization of **database deadlocks** .

Table candidates for concurrency control are identified during the "Data Modeling Exercise".

The only table which is updated concurrently is the Optimistic Locking mechanism. Once these are identified, the following is added to the application.

Add "N-Last
Updt" field to table in **database** ;
Error Handling routines on those operations which modify or delete from this table; and Display[Notification to user that the error has occurred.

96
Usage
The chart below describes the roles of the two basic **types** of components to enable optimistic **locking** .

Assumption: The optimistic **locking** field is of **type** Date and is named "N-Last-Updt"
Client Components Server Components
Read Store N -Last
Updt value in the Retrieve data (Always including N
Last...

14/3,K/45 (Item 28 from file: 349)
DIALOG(R)File 349:PCT FULLTEXT

00542275 **Image available**
A METHOD RELATING TO THE HANDLING OF TRANSACTIONS IN DATABASES
PROCEDE PERMETTANT DE GERER DES TRANSACTIONS DANS DES BASES DE DONNEES
Patent Applicant/Assignee:
TELEFONAKTIEBOLAGET LM ERICSSON,
Inventor(s):
RONSTRoM Mikael,
Patent and Priority Information (Country, Number, Date):
Patent: WO 200005648 A1 20000203 (WO 0005648)
Application: WO 99SE1301 19990722 (PCT/WO SE9901301)
Priority Application: SE 982598 19980722
Designated States:
(Protection type is "patent" unless otherwise stated - for applications prior to 2004)
AE AL AM AT AU AZ BA BB BG BR BY CA CH CN CU CZ DE DK EE ES FI GB GD GE GH GM HR HU ID IL IN IS JP KE KG KP KR KZ LC LK LR LS LT LU LV MD MG MK MN MW MX NO NZ PL PT RO RU SD SE SG SI SK SL TJ TM TR TT UA UG UZ VN YU ZA ZW GH GM KE LS MW SD SL SZ UG ZW AM AZ BY KG KZ MD RU TJ TM AT BE CH CY DE DK ES FI FR GB GR IE IT LU MC NL PT SE BF BJ CF CG CI CM GA GN GW ML MR NE SN TD TG
Publication Language: English
Fulltext Word Count: 8288

Main International Patent Class: **G06F-009/46**
International Patent Class: **G06F-017/30**
Fulltext Availability:
Detailed Description
Claims

Detailed Description
... TRANSACTIONS IN
DATABASES
FIELD OF INVENTION
The present invention relates to a method of handling the risk of deadlock-between mutually simultaneous transactions in a **database** .

DESCRIPTION OF THE BACKGROUND ART
When handling transactions in databases, it has long been known to allow a transaction that has been denied access to...

...two
(or more) common objects, then the risk of deadlock has been detected.

This may not necessarily mean that the situation will develop into a **deadlock** , since different time aspects with respect to when the different objects will be accessed by respective transactions, and the **type** of **lock** that is set on respective transactions on said different objects will decide whether or not the situation will develop into a **deadlock** situation.

The time aspect implies that a waiting time may depend on reasons other than a given transaction having caused a **deadlock** , such as a transaction must await access because of a **deadlock** caused by completely different transactions, or because of the time taken to carry out preceding transactions in the queue, or because the own access takes time to carry out, without resulting in **deadlock** ,
one **type** of **lock** enables access to be achieved as a purely reading transaction, wherewith this transaction will set a so-called divided lock on the object and the object will therefore still be available for reading other transactions.

When the danger of a **deadlock** is detected, the danger can be avoided by executing different procedures.

These procedures can be divided mainly into two groups, **deadlock** detection and **deadlock** avoidance.

When **deadlock** is detected there is initiated a process in which it is established whether or not **deadlock** is actually possible, prior to one of the involved transactions being aborted,
A **deadlock** is avoided by aborting one of the involved transactions immediately the risk of deadlock is detected, without actually establishing whether or not a deadlock exists...

...physically in another order in practice but interlinked in the consecutive order, for instance via a linked list.

For instance, when all tables in a **database** are given different numbers, table 1, table 2, table 3 and so on, all objects in the total **database** can be considered to lie in a number order relative to each otherf in other words in the consecutive order in the table in combination with the table order.

RECTM SHEET (RULE 91)
A **database** organisation of this **kind** enables **deadlock** between two transactions to be prevented, and therewith completely avoided.

This is possible by allowing solely those transactions that are active or ongoing in the **database** to access relevant objects in number order. If such is the case, a first transaction can never lock a first object and then "back out...is a non-strictly growing
transaction, but never between two strictly growing transactions.

It can also be mentioned that the difficulties encountered with monitoring a **database** with respect to deadlock are particularly significant in distributed **databases** .

The reader is referred to Principles of Distributed **Database** Systems by M, Tamer bzsu and Patrick Valduriez, ISBN 0 691643-0, for further information concerning **database** handling in general and **deadlock** and the handling of **deadlock** situations in particular.

19 A method according to Claim 18, characterised in that said categories include combinations of simple and complex transactions, reading...

14/3,K/46 (Item 29 from file: 349)
DIALOG(R)File 349:PCT FULLTEXT

00506743 **Image available**
DATABASE MANAGEMENT SYSTEM AND METHOD FOR CONDITIONAL CONFLICT SERIALIZABILITY OF TRANSACTIONS AND FOR COMBINING META-DATA OF VARYING DEGREES OF RELIABILITY
SYSTEME ET PROCEDE DE GESTION DE BASE DE DONNEES SERVANT A SERIALISER L'INCOMPATIBILITE CONDITIONNELLE DE TRANSACTIONS ET A COMBINER DES METADONNEES PRESENTANT DIFFERENTS DEGRES DE FIABILITE
Patent Applicant/Assignee:
TELENOR AS,
Inventor(s):
ANFINDSEN Ole J,
Patent and Priority Information (Country, Number, Date):
Patent: WO 9938095 A1 19990729
Application: WO 99NO18 19990125 (PCT/WO NO9900018)
Priority Application: US 9813678 19980126; US 9813808 19980126
Designated States:
(Protection type is "patent" unless otherwise stated - for applications prior to 2004)
AU CA JP NO AT BE CH CY DE DK ES FI FR GB GR IE IT LU MC NL PT SE
Publication Language: English
Fulltext Word Count: 15278
Main International Patent Class: **G06F-017/30**
Fulltext Availability:
Detailed Description

Detailed Description
... The both acquire IW locks on that table (and probably also on some higher level resources, such as file or tablespace, as well as the **database**), and then R, U or W locks on individual tuples. As long as 1 5 the two transactions do not access the same tuple, then there will no conflict. Should they happen to access the same tuple, then there will be a conflict, and the possibility of **deadlock** cannot be completely excluded. However, the potential **deadlocks** caused by overlapping IW requests are, in general, no worse than the potential **deadlocks** associated with other resource locking situations.

The complete lock (access mode) compatibility matrix is shown in Table 3.

B IR R U 1W RIW W...

...the intersection of the R and IW rows/columns.

In practice, browse (B) locks and exclusive (X) locks are not used at the lowest resource **levels** .

Using B **locks** at the lowest **levels** would, at least partially, defeat the purpose of using isolation level UR in the first place. For example, in a relational **DBMS** a transaction using the UR isolation level may request a browse lock at the table level (and all levels above the table level), and then...

14/3,K/48 (Item 31 from file: 349)
DIALOG(R)File 349:PCT FULLTEXT

00236234
METHOD AND APPARATUS EMPLOYING LOOKAHEAD TO REDUCE MEMORY BANK CONTENTION FOR DECOUPLED OPERAND REFERENCES

PROCEDE ET APPAREIL UTILISANT UNE RECHERCHE DE LECTURE PAR ANTICIPATION AFIN DE REDUIRE L'ENCOMBREMENT DES BLOCS DE MEMOIRE POUR DES REFERENCES D'OPERANDES DECOUPLEES

Patent Applicant/Assignee:
BIRD Peter L,

Inventor(s):
BIRD Peter L,

Patent and Priority Information (Country, Number, Date):
Patent: WO 9310496 A1 19930527
Application: WO 92US9534 19921105 (PCT/WO US9209534)
Priority Application: US 91241 19911114

Designated States:
(Protection type is "patent" unless otherwise stated - for applications prior to 2004)
CA JP KR RU AT BE CH DE DK ES FR GB GR IE IT LU MC NL SE

Publication Language: English
Fulltext Word Count: 6515
Main International Patent Class: **G06F-012/00**
Fulltext Availability:
Detailed Description

Detailed Description
... before the second store of 'A's If the system ignores this situation and moves the second store onto the store address queue, then a **deadlock** condition might arise in the machine. A naive check for FLOW dependencies case, the machine will halt since the second store requires that the statement containing the load be executed.

The solution employed by the present inventive system to prevent this **kind** of **deadlock** is to block the input address stream whenever a FLOW dependency relationship is encountered.

Blocking the input stream upon FLOW dependencies, and the fundamental synchronization...

...floating-point processor (FP) 12. The AP 10 consists of an integer function unit 20 (capable of addition and multiplication) and a 3-port register **file** 22, AP 10 operations are assumed to take a single cycle to complete. A path (not shown) to memory f or the AP The floating ...

14/3,K/49 (Item 32 from file: 349)
DIALOG(R)File 349:PCT FULLTEXT

00202664 **Image available**
INTEGRATED SOFTWARE ARCHITECTURE FOR A HIGHLY PARALLEL MULTIPROCESSOR SYSTEM
ARCHITECTURE DE LOGICIELS INTEGREE POUR SYSTEME HAUTEMENT PARALLELE A PROCESSEURS MULTIPLES

Patent Applicant/Assignee:
SUPERCOMPUTER SYSTEMS LIMITED PARTNERSHIP,

Inventor(s):
SPIX George A,
WENGELSKI Diane M,
GAERTNER Gregory G,
BRUSSINO Giacomo G,
HESSEL Richard E,
BARKAI David M,
HAWKINSON Stuart W,
JOHNSON Mark D,
BURKE Jeremiah D,
THOMPSON Keith J,
CHEN Steve S,
OSLON Steven G,

STROUT Robert E II,
MASAMITSU Jon A,
COX David M,
RASBOLD James C,
CRAMER Timothy J,
VAN DYKE Don A,
O'GARA Linda J,
O'HAIR Kelly T,
SEBERGER David A,
CHANDRAMOULI Ashok,

Patent and Priority Information (Country, Number, Date):
Patent: WO 9120033 A1 19911226
Application: WO 91US4066 19910610 (PCT/WO US9104066)
Priority Application: US 90466 19900611
Designated States:
(Protection type is "patent" unless otherwise stated - for applications prior to 2004)
AT AU BE CA CH DE DK ES FR GB GR IT JP KR LU NL SE
Publication Language: English
Fulltext Word Count: 27745

Main International Patent Class: **G06F-009/00**
Fulltext Availability:
Detailed Description

Detailed Description
... the resource manager to make sure
that each user gets what they want. And to so do in such a manner that
a
guarantees no **deadlocks** . Other **kinds** of jobs, jobs that do not
require
resources above a certain level, will run in the normal System V manner.

1 3 Resource Categories
Private...devices. Semi-private
Resources are those such as optical disks and drives, and compact disks.

Public Resources are fully shared. Examples include processors, System V
file systems, disks, main memory, secondary memory, and input/output
channels. The resource manager is concerned primarily with non-shared
resources,
1 4 Resource Management
A resource **database** map can be allocated dynamically. The resource
database is alterable on the fly, and this bit-wise database map that is